DECEMBER 1977 VOLUME 2, Number 12

BYTE

the small systems journal



$2.00 in USA



Your computer system needn't cost a fortune.

Some computer kits include little more than a mother board and a front panel, and you pay extra for everything else you need to make an operating computer.

SWTPC doesn't do it that way, so you can get your Southwest Technical 6800 Computer up and running at a bargain cost compared with most other systems. It comes complete at $395 with features that cost you extra with many other systems.

The Extras You Get

These extras include 4K of random-access memory, a mini-operating system in read-only memory, and a serial control interface. They give you 1) a considerable amount of working memory for your programs, 2) capability through the mini-operating system to simply turn on power and enter programs without having to first load in a bootstrap loader, and 3) an interface for connecting a terminal and beginning to talk with your computer immediately.

Low-Cost Add-Ons

Now that you have a working computer, you'll probably want to add at least two features soon, more memory and interfaces for needed accessory equipment. Memory for our 6800 is another bargain. You can get 4K memory boards for just $100 and 8K boards for only $250.

Our interfaces cost little compared with many other systems. For just $35 you can add either a serial or parallel interface board. (And you won't have to buy several interfaces on a costly board to get just the one you want.)

Peripheral Bargains

Your computer is no good without at least a terminal for entering data and viewing computer output, and you will probably want a good method of storing programs and data.

We offer you a line of high-quality peripherals at low prices. (You can prove this by just comparing prices.)

Buy our CT-64 Video Terminal for only $325 and our CT-VM Monitor with matching cover for $175. Our MF-68 Dual Minifloppy costs just $995, complete with Disk BASIC and a disk operating system. For cassette storage our AC-30 Cassette Interface gives simple control for one or two cassette recorders.

You can get inexpensive hard copy with our PR-40 Alphanumeric Line Printer.

We back up the 6800 system with low-cost software, including 4K and 8K BASIC.

Compare the value you get with our computer and peripherals before you buy. We think you'll find that SWTPC gives you more for your money in every way.

Enclosed is:

_____ $995 for the Dual Minifloppy
_____ $325 for the CT-64 Terminal
_____ $175 for the CT-VM Monitor
_____ $395 for the 4K 6800 Computer
_____ $250 for the PR-40 Line Printer
_____ $79.50 for AC-30 Cassette Inferface
_____ Or BAC # __________ Exp. Date_____
_____ Or MC # __________ Exp. Date_____

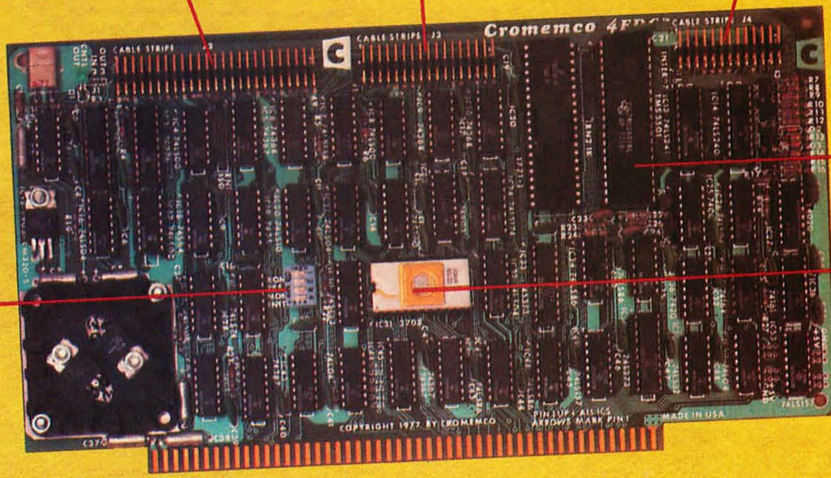
Name __________ Address __________

City __________ State __________ Zip __________

Southwest Technical Products Corp.
219 W. Rhapsody, San Antonio, Texas 78216
London: Southwest Technical Products Co., Ltd.
Tokyo: Southwest Technical Products Corp./Japan

Circle 109 on inquiry card.

The easy way to get disk storage, FORTRAN IV, and other programming power



Here's a new disk controller and disk drive combination that will set you up for truly powerful disk storage.

The new controller is extremely versatile. You can use it with **either** our **new 5″ single disk drive** or our **8″ dual disk drive.** In fact, the controller will **interface up to three** 5″ or **four** 8″ drives.

That means you can have enormous disk storage since the new controller puts 92 kilobytes on each side of a 5″ diskette and 256 kilobytes on an 8″ diskette. Recording is in soft-sectored IBM format.

FORTRAN IV AND MORE

You can get still more Cromemco disk operation aids. For example, we also offer FORTRAN IV for our computer users.

And as in so many things, we are the first manufacturer in the field to offer this advanced program for the Z-80 μP.

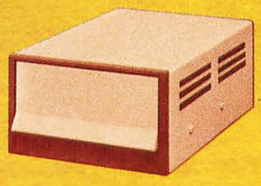
Besides FORTRAN IV we also offer our special BASIC (14-digit precision), our Z-80 Assembler, and now an entertainment diskette with over a dozen of our Dazzler® games.

KEYBOARD CONTROL

The new Model 4FDC disk controller (supplied in our Z-2D) is for our Z-2 computer or any S-100 bus computer using our Z-80 CPU card.

You should also know about these other capabilities of the new controller:

- Its PROM-resident Disk Operating System (RDOS) gives you keyboard control of your disk drive and also includes a bootstrap to load our powerful CDOS disk operating system supplied on all Cromemco diskettes.
- The controller will **interface your CRT terminal** through its RS-232 serial port. May save you an I/O.
- It has 5 programmable interval timers.
- It has vectored interrupts.
- And it has an 8-bit parallel input port and an 8-bit parallel output port.



Single 5″ disk drive



Dual 8″ disk drive



LOOK TO THE FUTURE

This new disk controller equips you for the future as well as for now. Not only can you now have very large storage, but the features of the controller and the standard IBM format protect you from early obsolescence.

STORES/FACTORY

This new card and the disk drives are in production and available.

So contact your computer store or the factory today and you can have the power of FORTRAN IV and a large memory right away.

PRICES

Model 4FDC-K Disk Controller kit . . . . $ 395
Model 4FDC-W Disk Controller assembled . . . $ 595
Model WFD 5″ single disk drive assembled . . $ 495
Model PFD-K 8″ dual disk drive kit $1995
Model PFD-W 8″ dual disk drive assembled . . $2495

Disk drives are complete with power supply, case and cables.

SOFTWARE

Purchasers of Cromemco computers or drives may purchase software on 5″ or 8″ diskettes as follows:

	5″ Diskette Model	8″ Diskette Model	Price
FORTRAN IV	FDF-S	FDF-L	$95
Z-80 Assembler	FDA-S	FDA-L	$95
16K BASIC	FDB-S	FDB-L	$95
Dazzler® games	FDG-S	FDG-L	$95

Cromemco
i n c o r p o r a t e d
Specialists in computers and peripherals
2400 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

Circle 34 on inquiry card.

You can now have the industry's finest microcomputer with that all-important disk drive



The professional-grade microcomputer for professionals

Shown with optional bench cabinet

YOU CAN GET THAT ALL-IMPORTANT SOFTWARE, TOO

Loading your programs and files will take you only a few seconds with the new Cromemco Z-2D computer.

You can load fast because the Z-2D comes equipped with a 5″ floppy disk drive and controller. Each diskette will store up to 92 kilobytes.

Diskettes will also store your programs inexpensively—much more so than with ROMs. And ever so much more conveniently than with cassettes or paper tape.

The Z-2D itself is our fast, rugged, professional-grade Z-2 computer equipped with disk drive and controller. You can get the Z-2D with either single or dual drives (dual shown in photo).

CROMEMCO HAS THE SOFTWARE

You can rely on this: Cromemco is committed to supplying quality software support.

For example, here's what's now available for our Z-2D users:

CROMEMCO FORTRAN IV COMPILER: a well-developed and powerful FORTRAN that's ideal for scientific use. Produces optimized, relocatable Z-80 object code.

CROMEMCO 16K DISK BASIC: a powerful pre-compiling interpreter with 14-digit precision and powerful I/O handling capabilities. Particularly suited to business applications.

CROMEMCO Z-80 ASSEMBLER: a macro-assembler that produces relocatable object code. Uses standard Z-80 mnemonics.

ADVANCED CONTROLLER CARD

The new Z-2D is a professional system that gives you professional performance.

In the Z-2D you get our well-known 4-MHz CPU card, the proven Z-2 chassis with 21-slot motherboard and 30-amp power supply that can handle 21 cards and dual floppy drives with ease.

Then there's our new disk controller card with special features:

- Capability to handle up to 4 disk drives
- A disk bootstrap Monitor in a 1K 2708 PROM
- An RS-232 serial interface for interfacing your CRT terminal or teletype
- LSI disk controller circuitry

Z-2 USERS:
Your Z-2 was designed with the future in mind. It can be easily retrofitted with everything needed to convert to a Z-2D. Only $935 kit; or $1135 for assembled retrofit package.

We're able to put all of this including a UART for the CRT interface on just one card because we've taken the forward step of using LSI controller circuitry.

STORE/FACTORY

Contact your computer store or Cromemco factory now about the Z-2D. It's a real workhorse that you can put to professional or OEM use now.

Kit: Z-2D with 1 disk drive (Model Z2D-K) .... $1495.
Assembled: Z-2D fully assembled and tested (Model Z2D-W).... $2095.
Additional disk drive (Model Z2D-FDD) .... $495.

SOFTWARE

(On standard IBM-format soft-sectored mini diskettes)

16K BASIC (Model FDB-S)......... $95
FORTRAN IV (Model FDF-S)........ $95
Z-80 Assembler (Model FDA-S).... $95

Cromemco
incorporated
Specialists in computers and peripherals
2400 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

Circle 34 on inquiry card.

In the Queue

BYTE DECEMBER 1977
Volume 2 Number 12

Foreground

Background

Nucleus

BYTE is published monthly by BYTE Publications Inc, 70 Main St, Peterborough NH 03458. Address all mail except subscriptions to above address; phone (603) 924-7217. Address all editorial correspondence to the editor at the above address. Unacceptable manuscripts will be returned if accompanied by sufficient first class postage. Not responsible for lost manuscripts or photos. Opinions expressed by the authors are not necessarily those of BYTE. Address all subscriptions, change of address, Form 3579, and fulfillment complaints to BYTE Subscriptions, PO Box 590, Martinsville NJ 08836. Second class postage paid at Peterborough NH 03458 and at additional mailing offices—USPS Publication No. 102410. Subscriptions are $12 for one year, $22 for two years, and $32 for three years in the USA and its possessions. Effective Jan 1 1978: $15 for one year, $27 for two years, and $39 for three years. Add $5.50 per year for subscriptions to Canada and Mexico. Canadian second class registration no. 9321. $25 for a one year subscription by surface mail worldwide. Air delivery to selected areas at additional rates available upon request. $25 for a one year subscription by air delivery to Europe. Single copy price is $2.00 in the USA and its possessions, $2.40 in Canada and Mexico, and $3.50 in Europe, and $4.00 elsewhere. Foreign subscriptions and sales should be remitted in United States funds. Printed in United States of America.

In This BYTE

This month's cover is based on Kurt J Schmucker and Robert M Tarr's article, **The Computers Of Star Trek** (page 12). It is an appropriate topic for computer people, many of whom are science fiction aficionados, Trekkies, and users of the Force. The theme, interpreted by artist Robert Tinney, is: What would happen if the crew of the *Enterprise* visited a holographic museum of ancient technology that had an exhibit devoted to personal computing, circa 1977? Robert used Willard Nico and his 8080 based computer system with dual floppy disk, video terminal and DECwriter as models for the diorama. The cassette recorder, made obsolete by the disk drives, is shown unused.

The floppy disk can give your computer the extra storage power needed for many applications such as advanced music and voice synthesis, artificial intelligence and robotics. Find out more about the ubiquitous floppy in Ira Rampil's **A Floppy Disk Tutorial.**

Microprocessor operation code structure is sometimes incompletely documented, as is demonstrated in two articles: Gerry Wheeler's commentary on **Undocumented M6800 Instructions** and H T Gordon's commentary on **The XF and X7 Instructions of the MOS Technology 6502.** The effects of the undocumented op codes are interesting, even if you don't want to use them as part of normal coding practices.

In a neat combination of tutorial and practical information, Bill Struve's article **A $19 Music Interface (and Some Music Theory for Computer Nuts)** provides a way to generate square wave musical tones for four channels as a result of an investigation of the theory of harmony.

Transform your computer into a powerful 8 channel 3½ digit voltmeter. Steve Ciarcia shows you how in the latest installment of **Ciarcia's Circuit Cellar.** Let a BASIC program do all your calculations and get results that compare favorably with expensive digital voltmeters. Read **On a Test Equipment Diet? Try an 8 Channel DVM Cocktail!**

Once upon a time, Jack and the Machine Talked; now Jack and his friendly 6800 have moved onto better things like debugging the programs issued by the assembler described in an earlier article. Turn to **Jack and the Machine Debug** by Grappel and Hemenway for a humorous (but tutorial) account of the development of a program called Tracer 6800 which uses software breakpoint techniques to provide an instruction by instruction machine code execution trace on a terminal or hard copy device.

To write well conceived programs easily, you have to design them in a disciplined and structured fashion. David A Higgins begins describing one useful method in the article on **Structured Programming with Warnier-Orr Diagrams, Part 1: Design Methodology.**

As a second installment in a series of articles, Stephen P Smith turns to the problems of motion in which effects of the motion's current state feed back into the model. Turn to **Simulation of Motion: An Automobile Suspension** for a more detailed model which features damping (shock absorbers) and bounce (springs) in response to external conditions (bumps in a road).

The use of interrupts allows you to keep track of several devices at the same time. If you are not familiar with the use of interrupts read Robert Wier's article, **A Little Bit on Interrupts.**

Constructing and interfacing a PolyMorphics Video Interface is described by Wayne Wenzlaff. Wayne describes his experiences with his video interface and how he modified a television set for use as a monitor in **Using the PolyMorphics Video Interface.**

Multiprogramming allows your computer to seemingly perform several tasks at the same time. It can save processor time by always having a program executing while another program waits for some type of input. Prof Irwin Lahasky's article, **Multiprogramming Simplified,** explains the basics of multiprogramming.

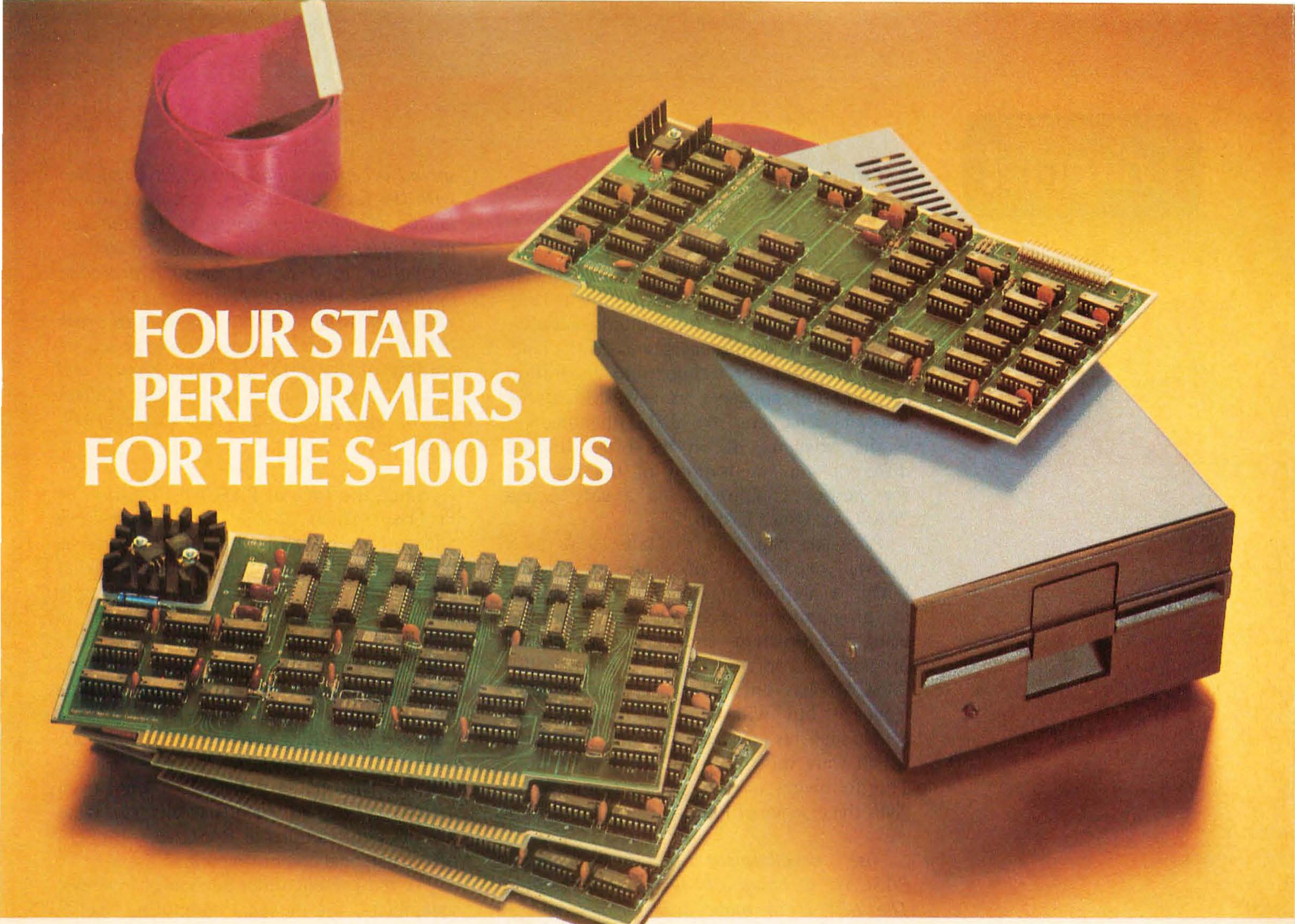
Many experimenters, including the editors of this magazine, have discovered the real advantages of purchasing used but eminently usable gear. Sol Libes gives valuable pointers to frugal hackers in **Where to Get Bargains in Used Computer Equipment.**

As personal computer users acquire more and more memory for their processors, thoughts can be turned to more powerful languages for the expression of programs. Gary McGath feels that small computer users should have nonnumeric, symbolic data manipulation abilities in their langusages. In **A Look at LISP,** Gary describes one of the candidates for such symbolic manipulations in the small computer.

Relative addressing allows jumps within a program to be made independent of the location of the program in memory address space. But what about such position independent code in processors like the 8080 which have no relative branch addressing? Read James P Gaskell's **Relative Addressing for the 8080** and learn how to simulate this feature for the 8080.

Handshaking is the process of coordinating two asynchronous processes, such as serial communication operations and a program. In a short article, Thomas McGahee shows how to **Save Software: Use a UART for Serial IO.**

What do you do if you're an oceanographer and want a microprocessor to help collect data at the bottom of the sea for eight weeks? One solution is to use a watertight titanium sphere and a battery powered processor. Henry Lahore shows how he did it in **A User's Report on Intercept Jr.**



Meet The North Star Family

THE NORTH STAR S-100 FAMILY—four high performance products at attractive low prices. Our boards are designed for use in the North Star HORIZON computer and other S-100 bus computers using 8080 or Z80 processors. Visit your computer store for a demonstration, or write for our free color catalog.

16K RAM BOARD

No other S-100 bus memory can match the performance of the North Star 16K RAM at any price. This low-power board has been designed to work at full speed (no wait states), even at 4MHz with both Z80 and 8080 systems. Memory refresh is invisible to the processor, bank switching is provided and addressability is switch selectable in two 8K sections. Best of all, a parity check option is available. Kit: $399. Assembled: $459. Parity Option — kit: $39. Assembled: $59.

MICRO DISK SYSTEM

The North Star MDS is a complete floppy disk system with *all* hardware and software needed to add floppy disk memory and a powerful disk BASIC to S-100 bus computers. The North Star MDS is widely considered one of the best designed and most complete S-100 bus products available. The MDS includes the S-100 interface board with on-board PROM for system startup, Shugart minifloppy disk drive, cabling and connectors, and DOS and BASIC software on diskette. Kit: $699. Assembled: $799. Additional drive — Kit: $400. Assembled: $450. Single Drive Cabinet: $39. Optional Power Supply: $39.

Z80A PROCESSOR BOARD

The North Star ZPB brings the full speed, 4MHz Z80A microprocessor to the S-100 bus. Execution is more than twice the speed of an 8080, and the ZPB operates in systems both with and without front panels. The ZPB includes vectored interrupts, auto-jump startup, and space for 1K of on-board EPROM. Kit: $199. Assembled: $259. EPROM Option — kit: $49. Assembled: $69.

HARDWARE FLOATING POINT BOARD

If you do number crunching, then this board is for you. The FPB performs high-speed floating point add, subtract, multiply and divide with selectable precision up to 14 decimal digits. Arithmetic is up to 50 times faster than 8080 software, and BASIC programs can execute up to 10 times faster. A version of North Star BASIC is included. Kit: $259. Assembled: $359. Prices subject to change.

NORTH STAR ★ COMPUTERS

2547 Ninth Street • Berkeley, California 94710 • (415) 549-0858

Circle 83 on inquiry card.

Editorial

Is PASCAL the Next BASIC?

By Carl Helmers

One of the most interesting phenomena in the academic world of computer science of late is the language PASCAL. This language is the subject of much intense activity, and is rapidly gaining acceptance as the language of choice for training and illustration of computer concepts to new students of the field. Characteristic of this phenomenon is the existence of on the order of 100 different implementations of the language for various computers and a very active "PASCAL User's Group."

PASCAL began in the late 1960s as a tutorial experiment of Professor Niklaus Wirth: a method of teaching the concepts of programming in a systematic fashion using a consistent and highly structured program representation. Historically, PASCAL has antecedents in the ALGOL language but with the addition of concepts such as record and file structures which were missing in ALGOL's definition. The following passage by Professor Wirth gives the essence of PASCAL's purposes. . .

The development of the language PASCAL is based on two principal aims. The first is to make available a language suitable to teach programming as a systematic discipline based on certain fundamental concepts clearly and naturally reflected by the language. The second is to develop implementations of this language which are both reliable and efficient on presently available computers.

The desire for a new language for the purpose of teaching programming is due to my dissatisfaction with the presently used major languages whose features and constructs too often cannot be explained logically and convincingly and which too often defy systematic reasoning. Along with this dissatisfaction goes my conviction that the language in which the student is taught to express his ideas profoundly influences his habits of thought and invention, and that the disorder governing these languages directly imposes itself into the programming style of the students.

There is of course plenty of reason to be cautious with the introduction of yet another programming language, and the objection against teaching programming in a language which is not widely used and accepted has undoubtedly some justification, at least based on short term commercial reasoning. However, the choice of a language for teaching based on its widespread acceptance and availability, together with the fact that the language most widely taught is thereafter going to be the one most widely used, forms the safest recipe for stagnation in a subject of such profound pedagogical influence. I consider it therefore well worthwhile to make an effort to break this vicious circle. [Quoted from the second edition of the *PASCAL User Manual and Report*, by Kathleen Jensen and Niklaus Wirth, Springer Verlag, New York, 1974, page 133.]

Since the time of PASCAL's creation by Professor Wirth, the language has become widespread, primarily because his tutorial purposes also happen to coincide with what one might want in a systems and applications programming language used in software development. In fact acceptance has been sufficiently widespread that there now exist implementations for some of the more common microprocessors in the personal computing field (using the *PASCAL User's Group Newsletter* as a source for this information in a listing of implementations in issue #8 recently published). What are the ramifications of PASCAL as it might affect personal computing users?

At the present time, outside of low level assemblers, the personal computing field is dominated by one language, BASIC. It is the high level language of choice for users of the equipment and for manufacturers who sell to the users of the equipment. Any attempted personal computing system design these days must come up to the standards of a reasonable BASIC (such as the Microsoft BASIC used by MITS, OSI, Commodore and others) or it will be at a relative disadvantage in the marketplace. This dominance of BASIC as a language is a fact of life in this field. A decade and a half of language design evolution has occurred since BASIC first came on the scene, yet it still dominates at the user level. Why?

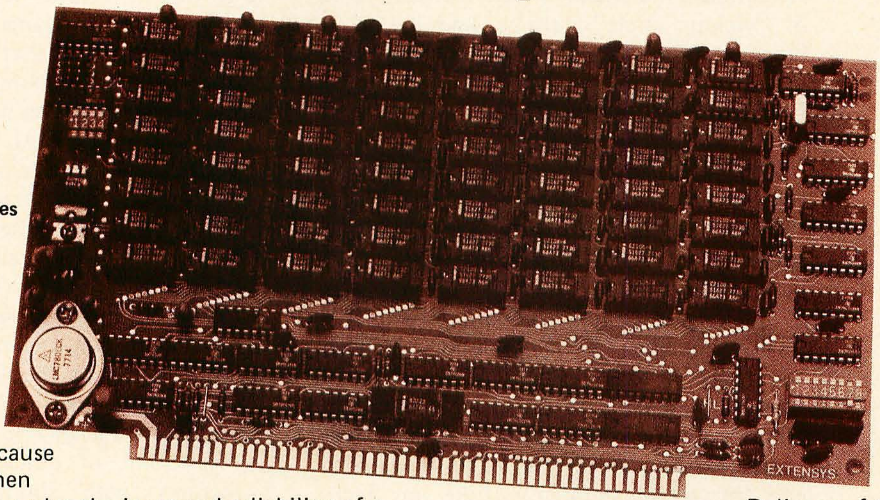
In a casual enumeration mode, I can list several fairly obvious and interrelated reasons why this has become the case; out of these reasons will come a similar scenario for development of PASCAL as a future option for personal computers.

- Everybody knows BASIC.
- BASIC has a manufacturer independent standard definition.
- Lots of implementations of BASIC are available.
- Much personal use applications software already exists in BASIC.
- BASIC is friendly.

extensys corporation

64K for $1495

The most cost effective products for your microcomputer.



RM64
64K bytes

THE EXTENSYS RM64 MEMORY BOARD provides the most cost effective system memory found in the industry. The RM64 provides this because of our low cost per byte when compared to our competition plus the increased reliability of a single board over multiple boards containing less memory. The board is S-100 bus compatible making it usable in over a dozen different microcomputer systems including ALTAIR and IMSAI. The RM64 is available in four configurations: 16K, 32K, 48K, or 64K bytes of memory all on ONE board. The board is completely assembled, checked out and burned in for over 100 hours prior to shipment. This complete testing procedure allows Extensys to provide a one year warranty on parts, labor and materials (assuming no misuse of the board occurs).

On board hardware is provided for:

- **Individual memory bank address selection in 8K byte increments;**
- **Complete dynamic refresh logic without loss of processing efficiency while programs are running;**
- **Board select logic which allows more than one 64K byte board per system;**
- **S-100 bus compatibility including on-board voltage regulator;**
- **Memory overlap which allows memory sharing the same address space to coexist in the same system;**
- **Write protection in 16K blocks; and**
- **Fully socketed for 64K allowing 16K, 32K, and 48K versions to be upgraded at a later date.**

Delivery of the RM64 is 15 to 30 Days upon receipt of order. Prices for the RM64 include shipping and handling prepaid in the continental United States.

Extensys Corporation is also announcing several other new, highly cost effective products. The FOS100 Floppy Disk System is based upon the Extensys File I/O Board and incorporates either one or two dual PerSci Model 277 floppy disk drives. The FOS100 System also includes the Extensys Multiprocessor Operating System, EMOS. EMOS provided many large system capabilities, such as multi-processor, multi-user operation and individual file privacy based on user supplied passwords. The MM16 Memory Manager interfaces to RM64 memory boards to create a megabyte or more of memory and adds full DMA capability to the FOS100 Floppy Disk System.

Extensys Fully Assembled and Tested Prices

RM64-16	$ 595	FOS100-2	$2880
RM64-32	$ 895	FOS100-4	$4680
RM64-48	$1195		
RM64-64	$1495	MM16	$ 295
16K Upgrade	$ 375		

Circle 52 on inquiry card.

extensys corporation
380 Bernardo Avenue
Mountain View, CA
94040
(415) 969-6100

At a superficial level, these reasons are part of a self-sustaining loop of circular reasoning: Since BASIC is friendly, everybody wants to know BASIC; since so many people learn BASIC, there tend to be lots of implementations. Much software for applications has been written in BASIC. Since a manufacturer independent standard for BASIC exists, conversion of programs from one machine to another is simplified, thus making widely available software useful to people, and so on . . . ad infinitum. . . This is Professor Wirth's "vicious circle."

Like many similar conventions, BASIC has been bootstrapped into the public awareness over time, and has acquired a certain inertia of its own that will keep it going for years in the same way that FORTRAN seems to live forever. Let's examine the reasons in this list, and in so doing compare BASIC to PASCAL, a language which is quite possibly in an earlier stage of a similar bootstrap cycle and may indeed become a much demanded "language of choice" for the user community. Vicious circles can have positive aspects: it all depends on which circle one has established. A contention I make is that the same sort of "vicious circle" can be, and indeed is being established for the language PASCAL.

Everybody Knows BASIC.

BASIC historically was introduced at a time when "big" computers dominated the field, and there was a need to partition the activities of such computers into small individually oriented packages for purposes of making the "big" computer available to many people. This partitioning succeeded admirably: when professor X (or Y or Z) wanted to make real exercises in programming available to students, BASIC was frequently employed, due to its availability and interactive simplicity. Like any technology, BASIC did not start out in an "everybody knows" state, but it got that way through its early availability and no small push from pedagogues of computer science.

Today, the teachers of programming are tending to push PASCAL as the language of choice for teaching "good" programming concepts. The PASCAL User's Group is evidence of the number of academic people who support the ideas of Professor Wirth to the extent of implementing their own local PASCAL systems for educational purposes. (This is typically done using a number of techniques of machine independence conceived by early implementors of PASCAL for purposes of spreading its implementations.) One result of this availability is that PASCAL is becoming the tool of teaching programming concepts which Professor Wirth envisioned . . . and the beginnings of the "everybody knows" state for PASCAL are already evident.

BASIC Has a Manufacturer Independent Standard Definition.

This comment is nominally true of BASIC. Work is indeed in progress on an ANSI standard for BASIC, and there is of course the original Dartmouth College definition of BASIC. The fact that people are trying to define a standard form of BASIC, however, is a result of the fact that the implementations of BASIC have been somewhat subject to variations. In the personal computing world, there are numerous differences at a detail level between language extensions of various BASIC interpreters, some as basic as the variations in string and array handling in various forms of minicomputer BASIC.

BASIC language implementors are no different from implementors of a number of languages, often succumbing to the "wouldn't it be neat if" syndrome and throwing in features not part of the original definitions of the language. The hitch with such featurism is that if anyone uses the features, the programs written with the feature may no longer be portable.

Of course PASCAL would be no more immune to featurism on the part of implementors; at least that would be an obvious contention since there is no fundamental difference between people who implement BASIC and people who implement PASCAL. But before making such a statement, an examination for the motives of implementation featurism should be made. BASIC in its original definition is a very limited and parochial language, one which represents a viewpoint of quick implementation of programs with limited IO formatting, standard floating point operations, and no intent to service large or complicated applications. Thus, many of the "feature" temptations presented to BASIC implementors are a result of attempts to correct the deficiencies of BASIC by adding omitted items (for example, strings, implemented differently in various BASIC interpreters).

PASCAL, on the other hand, by having a definition which is more general in scope than BASIC (although by no means complicated to use in simple problems) helps cut down these "feature" temptations on the part of its implementors. One basic example of this slightly more general definition is in PASCAL's inclusion of extensible data

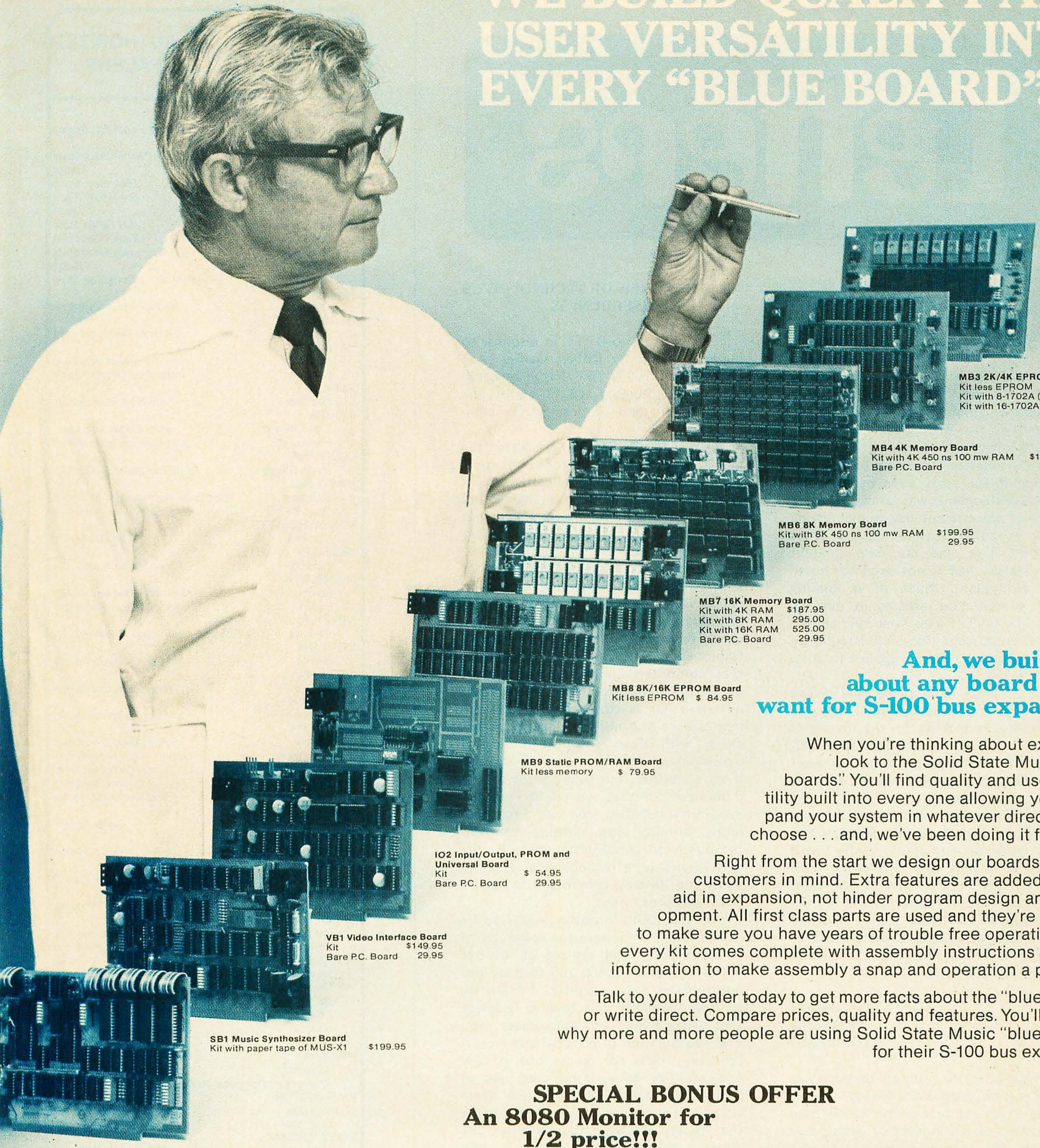
Continued on page 184

Articles Policy

BYTE is continually seeking quality manuscripts written by individuals who are applying personal computer systems, designing such systems, or who have knowledge which will prove useful to our readers. For a more formal description of procedures and requirements, potential authors should send a self-addressed, stamped envelope to BYTE Authors' Guide, 70 Main St, Peterborough NH 03458.

Articles which are accepted are purchased with a rate of up to $50 per magazine page, based on technical quality and suitability for BYTE's readership. Each month, the authors of the two leading articles in the reader poll (BYTE's Ongoing Monitor Box or "BOMB") are presented with bonus checks of $100 and $50. Unsolicited materials should be accompanied by full name and address, as well as return postage.■

WE BUILD QUALITY AND USER VERSATILITY INTO EVERY "BLUE BOARD"...



*Assembled boards also available at slightly higher prices.

And, we build just about any board you'll want for S-100 bus expansion.

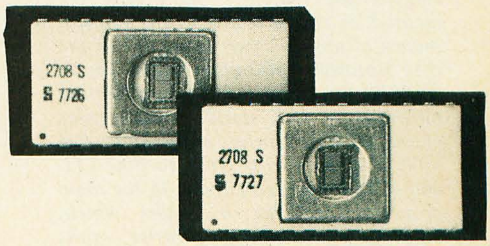
When you're thinking about expansion look to the Solid State Music "blue boards." You'll find quality and user versatility built into every one allowing you to expand your system in whatever direction you choose . . . and, we've been doing it for years.

Right from the start we design our boards with our customers in mind. Extra features are added that will aid in expansion, not hinder program design and development. All first class parts are used and they're checked to make sure you have years of trouble free operation. Plus, every kit comes complete with assembly instructions and user information to make assembly a snap and operation a pleasure.

Talk to your dealer today to get more facts about the "blue boards" or write direct. Compare prices, quality and features. You'll find out why more and more people are using Solid State Music "blue boards" for their S-100 bus expansion.

SPECIAL BONUS OFFER
An 8080 Monitor for 1/2 price!!!

If you buy any of the Solid State Music kits or assembled boards you'll receive a SSM8080 Monitor complete with either eight 1702's or two 2708's and over 50 pages of software information. A $49.95 retail value . . . just $25.00. Hurry!



cybercom

Solid State Music
2102A Walsh Avenue
Santa Clara, CA 95050
(408) 246-2707

Circle 106 on inquiry card.

We're the blue boards

Letters

DON'T KNOCK HOMEBREW

I just happened on Edgar Cohen's letter titled "Homebrew" (August 1977, page 12). He feels as I do, that we as readers should be looking at more homebrew computer articles in BYTE. My motive for writing was spurred by your follow-up comments.

We in amateur radio have been homebrewing it for years, even before commercial equipment was available. And much of the present day homebrewing is for economic reasons. At age 13, I wouldn't have had my own amateur radio station had it not been for home built equipment, much of it coming from construction articles in publications like BYTE. This same rule applies to test equipment I've built (frequency counter) and I think it applies to computers as well.

As a matter of fact, the only reason I don't have a computer yet is because practically everything on the market is out of my price range. To top if off, there are very few good surplus buys available, another way we in amateur radio get by cheaply.

I think you are wrong about the time invested in homebrew equipment. I and many other readers would gladly invest the time, if only the technology were there to give us a shove in the right direction. Certainly, the parts aren't expensive at all and much can be saved by wire wrapping instead of using ready to go printed circuit boards made up and sold by someone else.

Gary L Montgomery
WB3GBQ
Rockville MD

In reply, let me reiterate the fact that I am a confirmed homebrewer, always trying to get the best system by my own viewpoints on the subject of design. But perhaps the reason I don't view the method as cost effective is the remembrance of money spent to get one of the first Motorola 6800 chips ($360 for an XC6800 in early 1975) and similar purchase of 4 K static memory chips ahead of later price reductions. I've built a system with close to 800 TTL and MOS integrated circuits in the main processor modules, a system which, much to my homebrew surprise, works quite reliably. So I can certainly attest to the effectiveness of the approach. . .CH

ANOTHER VIEW OF TECHNOLOGY'S USEFULNESS

I am writing to BYTE in response to previous letters by Robert Garner (Ask BYTE, May 1977) and Nelson Ingersoll (Letters, September 1977) about the moral future and purpose of the computer.

In response to Mr Garner's letter, the real hazard that results from computers being "plugged into applications right and left" arises not from intangible and poorly defined moral abuses that might result from their introduction, but rather from the standpoint of cost effectiveness. Computers are being considered for such purposes as maintaining home food recipe files and keeping track of food inventory, where it is much simpler and cheaper to perform these tasks the old-fashioned way. I agree with Mr Garner that computers could be misused, but he seems to promote the same general hysteria that is currently plaguing the new science of genetic engineering. True, along with every new technological development comes a host of possibilities of misuse, but surely these can be minimized so that we could reap the maximum benefits from it? Sanctimonious morality does nothing but help the public to concentrate solely on the deficiencies of a technological development and destroy it before it has had a chance to prove itself.

Mr Ingersoll's problem seems to lie in the fact that he sees the computer as a device that will replace all of our duties as human beings and turn us into nonthinking, passive entities. I primarily see the computer as a tool, as a device that will aid man and *not* replace him. The "automation of physical and mental drudgery" does not relieve man of any responsibility to be productive and to think, but rather it frees him from simpler tasks (drudgery) in order to be the most creative and achieve the greatest self-satisfaction; man is only utilizing the feature that distinguishes him from the rest of the animal kingdom; in this instance, a decided advantage. I argue that performing simple and thoughtless tasks *is* drudgery. The "box" analogy is poor in that it presupposes it took no effort to design, build, and program the computer and robot to make the box. What Mr Ingersoll overlooks is that to accomplish this the man

Continued on page 19

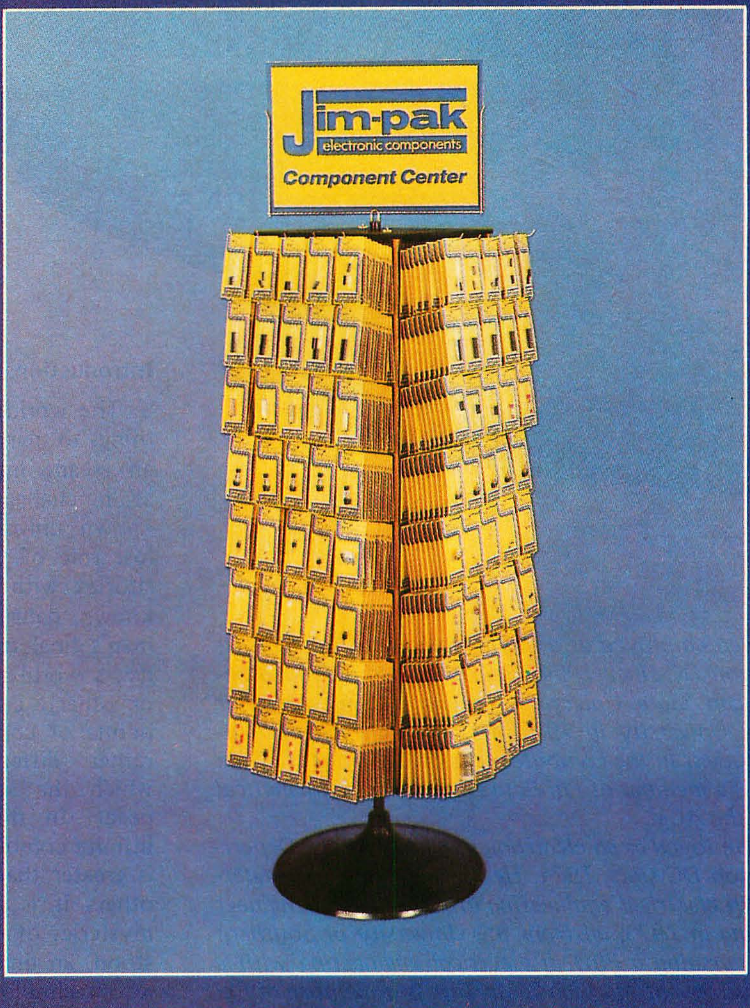
ATTENTION DEALERS:

Announcing

Jim-pak®

electronic components

One-Stop Component Center



- ★Over 200 quality items including integrated circuits, resistors, diodes, transistors, capacitors, connectors, switches, sockets, LEDs and Data Books covering all JIM-PAK® items.
- ★Immediate delivery on all orders
- ★Store display racks available
- ★Stock rotation and return policy
- ★Direct mail program available from list of active electronic buyers in dealers' area.
- ★National advertising campaign in leading electronics magazines to include list of qualifying dealers
- ★Nationally known manufacturers' products at prices every dealer can afford
- ★Guaranteed products
- ★Standard industry part numbers

A component line of proven sellers developed for the independent dealer. Ideal for computer shops, school stores, electronic dealers, hobby shops, or any location where there is a potential market for electronic sales.

A product line which supplies most of your needs from one distributor with a reputation for fast and efficient service. Attractive and compact display racks make initial installation of the JIM-PAK® line easy.

Your customers deserve the best. Now you can profitably retail name brand components at competitive prices. Be the first in your area to announce and sell the JIM-PAK® line. Write or call today.

FOR MORE INFORMATION AND PRICING SCHEDULE CONTACT:

Jim-pak® electronic components, a division of James Electronics, 1021 Howard Avenue, San Carlos, California 94070, (415) 592-8097

Circle 67 on inquiry card.

Kurt J Schmucker
Mathematician
Department of Defense
Washington DC

Robert M Tarr
Electronics Engineer
Department of Defense
Washington DC

The Computers of Star Trek



Introduction

The world of Star Trek represents many things to many people. To the majority it is an escape into a time when man is once again challenged by the vastness of his known universe and can assume the now lost role of the explorer of unknown territories with their inherent, but also unknown, dangers. To others it is the tale of man's final triumph over his own inhumanity; a world where race (human, humanoid or other), color and background are not points of contention and disunity, but are rather different reference points from which the society can grow together in peace. In this case "infinite diversity in infinite combinations" yields a whole which is greater than the sum of its parts. To yet others it is a time when science and the mysteries of nature are more clearly understood, an understanding which brings forth a technology beyond one's wildest imagination.

About the Authors

Kurt J Schmucker has been employed as a mathematician at the Department of Defense in Washington DC since 1974. He received an MS in mathematics in 1974 from Michigan State Univerisity, and an MS in computer science from Johns Hopkins University in 1977. Mr Schmucker is also currently an advanced special student in the computer science department of the University of Maryland. He is a member of Phi Beta Kappa, the Mathematical Association of America, and the ACM.

Robert M Tarr has been employed as an electronics engineer at the Department of Defense in Washington DC since 1974. He received an MS in materials science in 1972, an MS in electrical engineering in 1973, and an engineer degree in electrical engineering in 1975, all from the University of Southern California. He is currently pursuing a PhD in electrical engineering with a computers major at the University of Maryland. Mr Tarr is a member of Phi Beta Kappa, Eta Kappa Nu, and the IEEE.

Whatever view one takes of the Star Trek phenomenon, one can ask how closely our present culture approaches that ideal, be it in the area of science, human understanding, or man's view of himself in his world. This essay will try to measure one such gap between our own world and that of Star Trek: the state of computer technology.

Star Trek, with its fantastic starship the *Enterprise*, presents a level of computer science and computer engineering predicted by many science fiction writers.[1,2,3,4,5] Problems like voice IO, automatic programming in a natural language and computer analysis of complex, ill-defined problems are handled routinely. How close is the present technology to solving these problems? How soon can the computers of the *Enterprise* be constructed? Let us try to answer these questions.

Very little, if any, technical information is available on the *Enterprise* computers. Only vague, nontechnical statements are ever made by the Captain or the Science Officer.

> *"Deep in the heart of this ship are our computer banks. They operate the entire ship. They also contain the whole of human and humanoid knowledge. They are indisputably reliable. Our lives depend on them."*[6]
>
> *"In a matter of a few seconds we can obtain an answer to any factual question, regardless of its complexity."*[7]

No references are ever made to its physical configuration. Is it in fact one large central processor with terminals and perhaps terminal concentrators scattered throughout the ship, or a number of smaller optimized-for-special-functions processors loosely coupled through a shipwide network? The portions of the *Star Fleet Technical Manual* which recently have been made available to other than Star Fleet personnel do not include the "Ship's Computers Systems Schematics" section or the "Ship's Computers Maintenance Schematics" section.[8]

This essay will speculate on various hypotheses supported by the user level information available, and will attempt to show the hardware and software possibilities.

The Role of the Enterprise

The world of Star Trek takes place in the late 22nd or early 23rd century.[9,10] In that era the United Starship *Enterprise* is perhaps the second largest scientific and technical achievement resulting from the combined technologies of a unified portion of the Milky Way.[11] In the 22nd century a portion of the intelligent life forms of the galaxy have bonded together to form a union called the United Federation of Planets. It is a union more loose than the United States of America of the 19th, 20th and 21st centuries, yet stronger than the United Nations of the 20th century.[12] These planets occupy a significant portion of the Milky Way called the Treaty Exploration Territory, roughly a sphere centered on Sol with a radius of 4750 parsecs (approximately 15,500 light years). The Articles of the Federation were the agreement which established this union. They authorized funds for the building of a Star Fleet to act as the armed, peace keeping force of the Federation. The Fleet would be designed and built using state of the art techniques.

Included in this Star Fleet appropriation was an initial expenditure for 14 heavy cruiser starships, one to be named the *Enterprise.* These starships would be capable of extended duration patrol of the Treaty Exploration Territory and would be provided with weaponry and other capabilities enabling them to accomplish a myriad of possible tasks.

A starship on patrol represented the Federation in all matters within its occupied quadrant of space. Its functions included military uses (defense of borders and monitoring of intergalactic treaties with the Klingon and Romulan Empires, two other



"Computer, I want a complete rundown of all references to the design of computer systems for Enterprise class ships in the first quarter century of BYTE magazine."

"Yes, sir, but are you prepared for recursion?"

Federation-sized planetary unions, police actions (enforcement of Federation law, investigation of criminal actions), scientific missions (new explorations, data gathering assignments), diplomatic assignments, and missions of mercy.

It may be of benefit to compare the starship *Enterprise* to its namesake of the 20th century, the United States aircraft carrier *Enterprise*, in order to get a grasp of the scope involved in the construction of such a vessel. The *USS Enterprise*, launched in 1960, was the largest warship of its time and represented state of the art technology. It is still operational today and is powered by ten nuclear reactors. The *USS Enterprise* can sail on one set of reactor cores for about ten to 13 years or roughly 300,000 miles. It is provisioned every few weeks, however. The total cost of the construction and outfitting was 451 million dollars. In wartime, its complement is 5500 people consisting of 162 officers and approximately 2940 enlisted men plus 2400 airmen attached to an air wing.[13]

The starship *Enterprise*, on the other hand, is powered by a controlled matter-antimatter reaction.[14,15]

The total cost of the *Enterprise* was 50 billion credits.[16] On patrol, its complement consists entirely of 430 officers with 43 command (lieutenant and above) and 387 crew (ensign rank).[17] Traveling at the speed of light (ie: Warp Factor 1, approximately 1/200th of its flank speed) the *Enterprise* can travel for 18 years (as measured by a calendar traveling aboard the *Enterprise*?) without refueling or taking on additional provisions.[18]

Computer Uses and Capabilities of the Enterprise

An enterprise as complex as a starship requires a vast amount of computer support. Many of the routine tasks, such as monitoring of life-support systems, food synthesis and turbolift control (a vertical and horizontal elevator used for intraship transportation) are automatically maintained and controlled by the ship's computers. These are manually controlled only during an emergency.

Even more demanding and less well-defined tasks are routinely delegated to the ship's computers. The computers can activate the ship's alert systems and initiate the deflector shields (a set of very sophisticated defensive force shields which block matter and selective energy transmission) upon analysis from the ship's sensors. It appears that only a portion of the ship's vast amount of sensory data is routinely patched through the computer, since it is often instructed to "tie in to all ship's sensors" and provide the command staff with an analysis of unknown phenomena.[19,20,21]

If most of the *Enterprise* computing power resides in a central processing unit, this may point to a limitation in the computer's processing power. Perhaps if all data from all sensors were routinely processed there would be insufficient computing power to perform the remaining essential functions. If, on the other hand, there is a network of processors, then it may be that the results of sensor input are not routinely passed to the central complex.

Certain groups of sensors may be tied to local processors which perform the necessary functions, eg: activation of shields. The central processor complex does not routinely have need for these results unless correlation of the sensor data is required.

Computer control of the weapons systems (phasers and photon torpedoes) is also possible through analysis of sensor input, but it is not as accurate as computer assisted human control coupled with visual contact with the desired target.[22] This, too, may point to either a lack of sufficient computing power for timely analysis, or a lack of appropriate decision algorithms for this type of situation.

Another major use of the computer, perhaps the one most apparent to the casual observer, is information storage. The need for an immense data base can be clearly seen when one considers the role a starship plays in its patrol area. A starship captain acts autonomously from Star Fleet Headquarters in almost all aspects of his command.[23] As far as violations of Federation law are concerned, he is the judge and jury. In addition, the officers and crew often require instant access to the immense technical knowledge of the Federation in order to cope with new phenomena encountered in their patrols. This requires the existence of an immense interactive data base on the *Enterprise* itself. Because of the intergalactic distances involved, there can be no link to the computers at Star Fleet Headquarters or on Memory Alpha.[24] Subspace communications require time for transmission, too (often on the order of days).[25,26]

The vast amount of information stored in this data base is best appreciated by relating two incidents that happened on the *Enterprise*. The first involved a small space cruiser traveling toward Ophiuchus VI without an identification beacon. Upon pursuit, the cruiser entered an asteroid belt. After

Continued on page 172



Look To The North Star HORIZON Computer.

HORIZON™— a complete, high-performance microprocessor system with integrated floppy disk memory. HORIZON is attractive, professionally engineered, and ideal for business, educational and personal applications.

To begin programming in extended BASIC, merely add a CRT or hard-copy terminal. HORIZON-1 includes a Z80A processor, 16K RAM, minifloppy™ disk and 12-slot S-100 motherboard with serial terminal interface — all standard equipment.

WHAT ABOUT PERFORMANCE?

The Z80A processor operates at 4MHZ — double the power of the 8080. And our 16K RAM board lets the Z80A *execute at full speed.* HORIZON can load or save a 10K byte disk program in less than 2 seconds. Each diskette can store 90K bytes.

AND SOFTWARE, TOO

HORIZON includes the North Star Disk Operating System and full extended BASIC on diskette ready at power-on. Our BASIC, now in widespread use, has everything desired in a BASIC, including sequential and random disk files, formatted output, a powerful line editor, strings, machine language CALL and more.

EXPAND YOUR HORIZON

Also available— Hardware floating point board (FPB); additional 16K memory boards with parity option. Add a second disk drive and you have HORIZON-2. Economical serial and parallel I/O ports may be installed on the motherboard. Many widely available S-100 bus peripheral boards can be added to HORIZON.

QUALITY AT THE RIGHT PRICE

HORIZON processor board, RAM, FPB and MICRO DISK SYSTEM can be bought separately for either Z80 or 8080 S-100 bus systems.

HORIZON-1 $1599 kit; $1899 assembled.
HORIZON-2 $1999 kit; $2349 assembled.

16K RAM—$399 kit; $459 assembled; Parity option $39 kit; $59 assembled. FPB $259 kit; $359 assembled. Z80 board $199 kit; $259 assembled. Prices subject to change. HORIZON offered in choice of wood or blue metal cover at no extra charge.

Write for free color catalogue or visit your local computer store.

NORTH STAR ★ COMPUTERS

2547 Ninth Street • Berkeley, California 94710 • (415) 549-0858

Circle 83 on inquiry card.



Introducing Apple II.™
DOW JONES INDUSTRIAL AVERAGE= 933
TOTAL DAILY TRADING VOLUME= 25 SHARES

The home computer that's ready to work, play and grow with you.

Clear the kitchen table. Bring in the color T.V. Plug in your new Apple II,* and connect any standard cassette recorder/player. Now you're ready for an evening of discovery in the new world of personal computers.

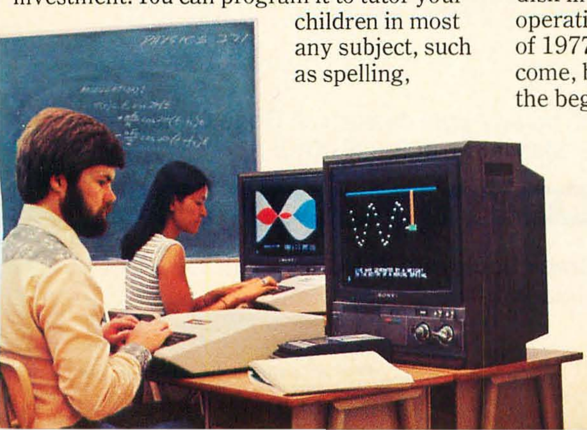
Only Apple II makes it that easy. It's a complete, ready to use computer—not a kit. At $1298, it includes features you won't find on other personal computers costing twice as much.



Features such as video graphics in 15 colors. And a built-in memory capacity of 8K bytes ROM and 4K bytes RAM—with room for lots more. But you don't even need to know a RAM from a ROM to use and enjoy Apple II. It's the first personal computer with a fast version of BASIC—the English-like programming language—permanently built in. That means you can begin running your Apple II the first evening, entering your own instructions and watching them work, even if you've had no previous computer experience.

The familiar typewriter-style keyboard makes communication easy. And your programs and data can be stored on (and retrieved from) audio cassettes, using the built-in cassette interface, so you can swap with other Apple II users. This and other peripherals—optional equipment on most personal computers, at hundreds of dollars extra cost—are *built into* Apple II. And it's designed to keep up with changing technology, to expand easily whenever you need it to.

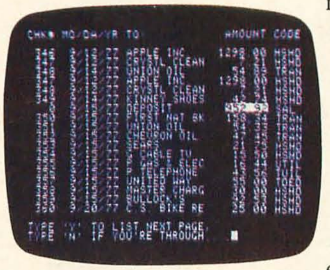
As an educational tool, Apple II is a sound investment. You can program it to tutor your children in most any subject, such as spelling, history or math. But the biggest benefit—no matter *how* you use Apple II—is that you and your family increase your familiarity with the computer itself. The more you experiment with it, the more you discover about its potential.



Start by playing PONG. Then invent your own games using the input keyboard, game paddles and built-in speaker. As you experiment you'll acquire new programming skills which will open up new ways to use your Apple II. You'll learn to "paint" dazzling color displays using the unique color graphics commands in Apple BASIC, and write programs to create beautiful kaleidoscopic designs.

As you master Apple BASIC, you'll be able to organize, index and store data on household finances, income tax, recipes, and record collections. You can learn to chart your biorhythms, balance your checking account, even control your home environment. Apple II will go as far as your imagination can take it.

Best of all, Apple II is designed to grow with you. As your skill and experience with computing increase, you may want to add new Apple peripherals. For example, a refined, more sophisticated BASIC language is being developed for advanced scientific and mathematical applications. And in addition to the built-in audio, video and game interfaces, there's room for eight plug-in options such as a prototyping board for experimenting with interfaces to other equipment; a serial board for connecting teletype, printer and other terminals; a parallel interface for communicating with a printer or another computer; an EPROM board for storing programs permanently; and a modem board communications interface. A floppy disk interface with software and complete operating systems will be available at the end of 1977. And there are many more options to come, because Apple II was designed from the beginning to accommodate increased power and capability as your requirements change.

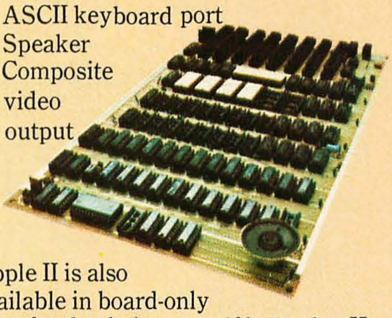


If you'd like to see for yourself how easy it is to use and enjoy Apple II, visit your local dealer for a demonstration and a copy of our detailed brochure. Or write Apple Computer Inc., 20863 Stevens Creek Blvd., Cupertino, California 95014.

Apple II™ is a completely self-contained computer system with BASIC in ROM, color graphics, ASCII keyboard, lightweight, efficient switching power supply and molded case. It is supplied with BASIC in ROM, up to 48K bytes of RAM, and with cassette tape, video and game I/O interfaces built-in. Also included are two game paddles and a demonstration cassette.

SPECIFICATIONS

- **Microprocessor:** 6502 (1 MHz).
- **Video Display:** Memory mapped, 5 modes—all Software-selectable:
  - Text—40 characters/line, 24 lines upper case.
  - Color graphics—40h x 48v, 15 colors
  - High-resolution graphics—280h x 192v; black, white, violet, green (16K RAM minimum required)
  - Both graphics modes can be selected to include 4 lines of text at the bottom of the display area.
  - Completely transparent memory access. All color generation done digitally.
- **Memory:** up to 48K bytes on-board RAM (4K supplied)
  - Uses either 4K or new 16K dynamic memory chips
  - Up to 12K ROM (8K supplied)
- **Software**
  - Fast extended Integer BASIC in ROM with color graphics commands
  - Extensive monitor in ROM
- **I/O**
  - 1500 bps cassette interface
  - 8-slot motherboard
  - Apple game I/O connector
  - ASCII keyboard port
  - Speaker
  - Composite video output



Apple II is also available in board-only form for the do-it-yourself hobbyist. Has all of the features of the Apple II system, but does not include case, keyboard, power supply or game paddles. $798.

PONG is a trademark of Atari Inc.

*Apple II plugs into any standard TV using an inexpensive modulator (not supplied).

Circle 4 on inquiry card.

apple computer inc.™

Continued from page 10

must have been extremely proficient at building boxes, for otherwise the programming would have been impossible. (Anyone who programs computers would know this.) To illustrate my concept of the computer as a tool, I propose that once the computer and robot have freed the man from the *drudgery* of making simple boxes, he can really think about what larger structure or scheme those boxes will be a part of. Following these guidelines will make the computer a safe and useful tool that benefits all society.

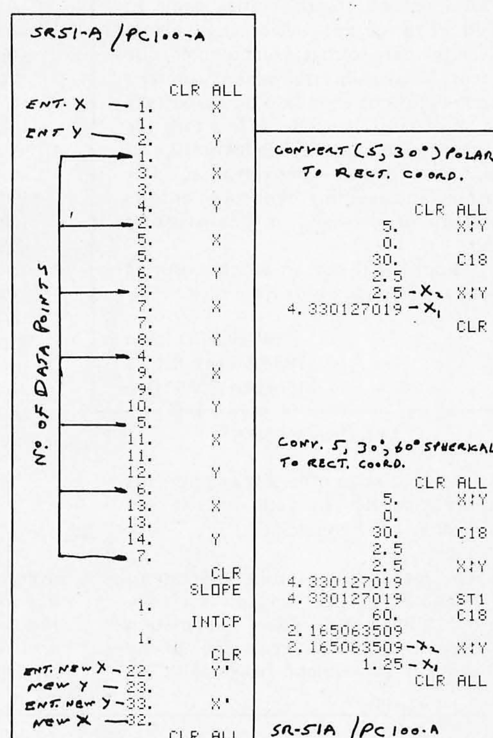
Mark Bizer
43 Morgan Cir
Amherst MA 01002

CONFIRMATION OF THE SR-51A PRINTER TRICK. . .AND SOME GRAPEVINE JUICE

After reading Webb Simmons' letter in the September 1977 BYTE regarding the use of his SR-51 TI calculator on the PC-100A printer, I decided to try the same with my SR-51A. Obviously this model functions on the PC-100A as well as the SR-51 as shown by the printout tapes I have included with my letter. The longer tape is a line regression problem which I "made up" on the spur of the moment in my haste to try out the SR-51A on the printer. Each 'x' value (1, 3, 5, 7, 9, 11, 13) and each 'y' value (2, 4, 6, 8, 10, 12, 14) is notated at the edge of the tape and the number of data points entered (7) are also shown. The shorter tape shows the printout of two problems I copied from the owner's manual of the SR-51A on pages 40 and 42 respectively. The first is the conversion of polar coordinates (5,30°) to rectangular coordinates while the second is a conversion of spherical (5, 30°, 60°) to rectangular coordinates.

In addition to the SR-51A I also own the SR-50, SR-56 and the TI Programmable 58 "Solidstate Software" and of course the PC100A printer. As I am a newcomer to the hobby of home calculator and computer systems I would appreciate hearing from other readers who use these calculators I have mentioned, in order to exchange ideas and information with them. I would like to know especially if anyone who practices chemistry or biochemistry as a hobby or as a student has any programs for these subjects that could be used or modified for use on the TI-58 or SR-56?

Finally, I have heard that TI intends to release a "programmer" for the TI-58/59 which will allow users to record their own library modules and also is planning on video monitors and XY plotters to add on to their programmable calculators. Are these idle



rumors or do you know if there is any truth to them?

William A Faria
74 Division St
New Bedford MA 02744

NEED NEW PLAYER PIANOS?

Since you didn't mention it in the September 1977 issue, let me be the first to tell you about Superscope's latest entry:

It's called the Pianocorder (a Superscope trademark), and it is an electronic player piano. Built around an 8080 processor, it's scheduled to come in two versions: a Vorsetzer, for grand pianos and such, and what seems to be a retrofit kit that can be put in an upright or spinet. This version is supposed to come out around $1500 or so. The control is by means of a digital casette tape, which controls which keys are down, and also controls the dynamics (by controlling the pulses into the solenoids). There are two things that are really exciting about this: there's a record mode (casually invite Van Cliburn over and have him try out your new piano); and the device will come with 100 cassettes, recorded from Mr Tuchinsky's collection. He's president of Superscope, and an avid fan of player piano rolls from Welte; his collection includes most of the great pianists from the turn of the century, who cut rolls for Mr Welte.

You can get more details from the source by contacting Tony Blazina, manager, Pianocorder Division, Superscope Inc, 20525 Nordhoff St, Chatsworth CA 91311. He says that basic descriptive literature should be available in a few weeks, and marketing early in 1978. I first read about it in a recent issue of *Electronics* magazine.

A long list of possibilities come to mind. The output device is a piano. Given the data format, you could encode a score, make an arrangement for four hands (why not eight?). You could take the sheet music, encode it, fine tune the phrasing, pedalling, dynamics until you had the perfect performance. For learning and teaching, you could encode the right or left hand, or one part of a duet.

I hope we'll see an article covering these bases not too long from now.

Michael D Zorn
1833 S Peck Rd #4
Monrovia CA 91016

APL COMMENTS

I would like to offer a few comments on APL matters in your August and September 1977 issues:

1. The readability of APL expressions might be brought out more clearly by adding to your explanation (September BYTE, page 166) of my thought experiment (August BYTE, page 40) this sequence (box below).
2. There are many serious errors in the APL references. For example, there are five errors in those on page 65 (August) alone. Perhaps you should consider publishing a carefully checked and annotated bibliography for APL.
3. In discussing APL, care should be taken to use appropriate terminology. For example, Mr Wimble's article confuses the important distinction between *operators* and *functions* (already commented upon in Mr Anthony's letter (August page 17)) and uses the term *arity* for *valence.* Perhaps the best reference for terminology and fundamental concepts is one not yet mentioned in your articles: *APL Language Manual,* available from IBM Corp as Form #GC26-3847.

Kenneth Iverson
163 Great Oak Ln
Pleasantville NY 10570

Mike Wimble concurs; the confusion over operators and functions is in part due to changes and revisions to APL over the years. We regret the bibliographic errors, and will attempt to correct them in future APL articles. ■

```
                      ⍳N  All candidates for primes to N.
               (⍳N)∘.|⍳N  A remainder table.
             0=(⍳N)∘.|⍳N  Divisibility table. Column I
                            shows the divisors of I.
         +/⍉0=(⍳N)∘.|⍳N  The number of divisors for each of ⍳N .
       2=+/⍉0=(⍳N)∘.|⍳N  A sieve showing which numbers have
                            exactly two divisors (the primes).
(2=+/⍉0=(⍳N)∘.|⍳N)/⍳N  Selection of the primes by the sieve.
```

PRIMER REMOVER

It was good to see so much attention to APL in the August 1977 issue. But you goofed when you recommended the *APL Primer.*

The *APL Primer* may be easy to read, but its author missed the boat on two key attributes of APL. He neglected APL's treatment of arrays as wholes, and its treatment of programs as "user defined functions" that take arguments, return results, and behave just like primitives. The truth is, he didn't realize at the time either how central those two topics are, or how easy to introduce they'd prove to be.

Experience shows that they're not hard to get across. By shrinking from forthright treatment of these fundamental aspects of APL (and thereby fostering dreadful programming habits), the *APL Primer* does the newcomer to APL a great disservice.

You shouldn't recommend it. IBM should withdraw it.

Paul Berry
I P Sharp Associates Inc
Ste 110, 299 California Av
Palo Alto CA 94306

PS: I know; I wrote it.

Rated G

The Best Game in Town.



Welcome to ComputerLand. An incredible adventure into the world of personal computers. A one-of-a-kind shopping experience.

Each ComputerLand store presents everything you ever wanted to know about computers. And then some.

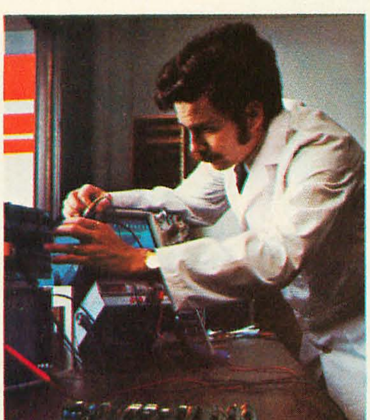
Take our Game Room, for starters. You'll find excitement for the whole family in our endless variety of challenging computer games. You can battle the Klingons in an out-of-this-world game of Star Trek. Create an electronic work of art with a computer controlled TV. Test your skill in a game of computerized hangman.

You can even plot your biorhythm.



But we're more than just fun and games. Each ComputerLand store offers a knowledgeable and personable staff of professionals to serve you.

Plus the greatest available selection of micro components. Whether it's a data processing system for your business or a computer controlled sprinkler system for your home, you'll find whatever you need at ComputerLand.



Read on.

Genuine Service.

We want to supply you with the one system that's right. Rather than a complete system that isn't. Or a limited system that is.

That's why, at ComputerLand, you deal with real professionals who are also real people. People who speak your language . . . in addition to BASIC, COBOL or FORTRAN.

People, in short, who can offer both the novice and the old hand the same expert guidance in selecting the optimum system he or she needs.

Yet, assisting in the purchase is only the beginning of ComputerLand's service. If the kit you bought requires a little more do-it-yourself than you yourself can do, we provide assembly assistance.

If that complex program proves to be just that, we provide programming assistance.

And if your system breaks down, our in-store service department will get you back up and running.

Right now!

Great Selection.

Your first stop at ComputerLand may well be your last stop.

ComputerLand offers the finest quality and largest selection of all the major brand names. Like Apple Computer, Cromemco, DEC, Diablo, Hazeltine, ICOM, IMSAI, Lear Siegler, National Semiconductor, North Star, Texas Instruments, Vector Graphics and more.

Plus a complete inventory of tools, books and accessories.

What's more, at ComputerLand, we deal in product. Not promises. Our inventory is on our own shelves. Rather than the manufacturer's. So you can take delivery on tomorrow's components today.



Which means, simply put, that at ComputerLand, you get exactly what you want.

Exactly when you want it.

Be Our Guest.

Begin with the grand tour of our exhibit areas. "Test-drive" any of our individual systems.

Then tell us your needs. We'll sit down and talk about the system that's right for you. It's as easy as that at ComputerLand.

The great computer store. RATED G.

Call or write for the address of the ComputerLand store nearest you. Franchise opportunities available.

ComputerLand™

1922 Republic Avenue, San Leandro, CA 94577 (415) 895-9363

Circle 30 on inquiry card.

.L THER W!

All Equinox software is designed to make full use of COPE capabilities. So loading your Equinox software is as simple as selecting an address and pressing RUN. Cassette data storage has never been more convenient.

The Equinox I/O Interface is programmed for the highly-reliable 300 baud Kansas City Standard for cassette data transfers. . .but can be reprogrammed for faster rates in the future. Your TTY/ RS-232 channel also features user-programmable rates up to 4800 baud. And you have access to all COPE routines.

There is no more cost-effective way to handle your system interfacing. And the Equinox I/O Interface is even more attractively priced when purchased with a complete system.

THE MEMORY

You can get your Equinox System TM together with more low-cost memory.

The Equinox System TM is the first system to offer the ECONORAM III TM 8K×8 memory board with the advanced SynchroFresh TM natural-timing refreshing system.

Configured as two individually addressable 4K blocks, the ECONORAM III TM comes assembled, tested and guaranteed to work perfectly in your Equinox System TM for one full year. It is also available as a lower-cost kit.

For tight budgets, the K-Ration TM 4K×8 static memory kit is also available for your Equinox System TM. This high-efficiency design is one of the most widely used, lowest power and lowest cost memories available for S-100 systems.

And the memory savings are even greater when you purchase your Equinox System TM all together!

THE SOFTWARE

The Equinox System TM now includes two primary software packages: EQU/ATE for text-oriented programming, editing and assembly, and BASIC-EQ language for number-oriented programming.

Both of these powerful programming languages have been greatly enhanced by full integration with our COPE interfacing software.

EQU/ATE gives you the power of a fully programmable text language. Using EQU/ATE, you can create programs to edit copy, keep inventory, manipulate your files of addresses or recipes, and more.

EQU/ATE is also one of the most powerful editor/assemblers available. EQU/ATE writes, assembles and executes your programs. It is fully symbolic, accepts global symbols and assembles programs larger than available memory.

EQU/ATE is also an extraordinary text editor which works interactively with you to edit by character, string or line. It also offers full number editing of memory contents in octal, hex or decimal.

Yet, EQU/ATE is extremely compact, occupying only 4K of memory with 8K total memory required.

BASIC-EQ is a powerful, interactive, high-level programming language developed for math and science applications. Its primary use is in the manipulation of numbers, although it is capable of string oriented operation. Our version, BASIC-EQ, occupies 5K of memory.

Both BASIC-EQ and EQU/ATE are supplied on cassette, ready to load themselves into your Equinox System TM. They are competitively priced. . .and are even more attractive when purchased together with your Equinox System TM.

GET IT TOGETHER!

For more information and ordering:

BY MAIL Send check or money order to Equinox Division, Parasitic Engineering, P.O. Box 6314, Albany, California 94706

BY PHONE 800-648-5311. BankAmericard/Visa and Master Charge accepted.

IN PERSON See The Equinox System TM at your local computer shop.

***A trademark of Teletype Corporation.**



X SYSTEM T.M.

A Floppy Disk Tutorial

Ira Rampil
917 Engineering Research Bldg
University of Wisconsin
Madison WI 53706

What peripheral device most often defines the home hacker's ultimate system? It is, of course, the floppy disk. But what are these devices that seem to have the ability to transform the smallest microprocessor system into a full-fledged computer? How do they work, and are they worth the cost? I slowly uncovered the answers to these questions as I sought to upgrade my system by adding floppies.

Basically, the floppy disk is the little cousin of IBM and other manufacturers' huge hard disk drives. As far as any computer is concerned, the floppy is a real disk drive. The differences between it and (for example) an IBM 3330 disk are mainly specifications of speed and storage capacity. Floppies, like other disks, are relatively fast random access memories. If the last three words sound familiar, it should be no surprise. Semiconductor random access memories store (if programmable) and read data by address, with a unit quantity of data (typically one bit) at each address. The data at any address can be quickly and easily changed without disturbing the contents of any other address.

So it is with floppy disks except that the access times are now measured in centiseconds instead of nanoseconds, and the quantity of data at each point is now hundreds or thousands of bits instead of just one. Because the structure of disk storage is similar to that of main memory, it is often used to store programs and data, especially those programs and data which are frequently referenced or modified. In fact, during the early age of electronic digital computers, machines like the IBM 650 actually used a rotating drum similar to a disk as its only memory, which fetched new instructions for execution with each revolution. It is widely believed that the Minuteman missile system still uses such a memory.

All of the other mass storage techniques available to hackers, such as paper tape, audio, and even digital cassettes, are fundamentally serial memories. That is, all or most of the recorded data may have to be passed through in order to find a particular piece of data.

Table 1 compares several different mass storage techniques. As you can see, floppy disks fall between hard surface cartridge

Table 1: A comparison of several different mass storage techniques.

Technique	IBM 2315 Cartridge Disk	IBM 3740 Floppy	Digital Cassette	Audio Cassette	Units
Data Capacity	48.	3.0	6.0	0.84	Million bits (unformatted)
Average or Typical Access Time	.035	.45	20	120*	Seconds (* = manually controlled)
Data Transfer Rate	2500	250	10.	0.3	k bps
Price of Commercial Package: Drive + Power + Controller	$8000	$1500	$1000**	$100	**Note that personal computing digital cassettes can be much cheaper than commercial drives
System Cost per Unit Data Rate	.32	.6	10	33.3	cents per bps
System Cost per Unit Storage	.016	.05	.016	.012	cents per bit stored
Media Cost	$100	$8.	$4.	$4.	(unit quantity prices)
Storage Cost	1.7	2.2	0.55	3.9	cents/kilobyte of media

Typetronic:
your source for complete computer management systems... now introduces New Jersey's most complete computer store

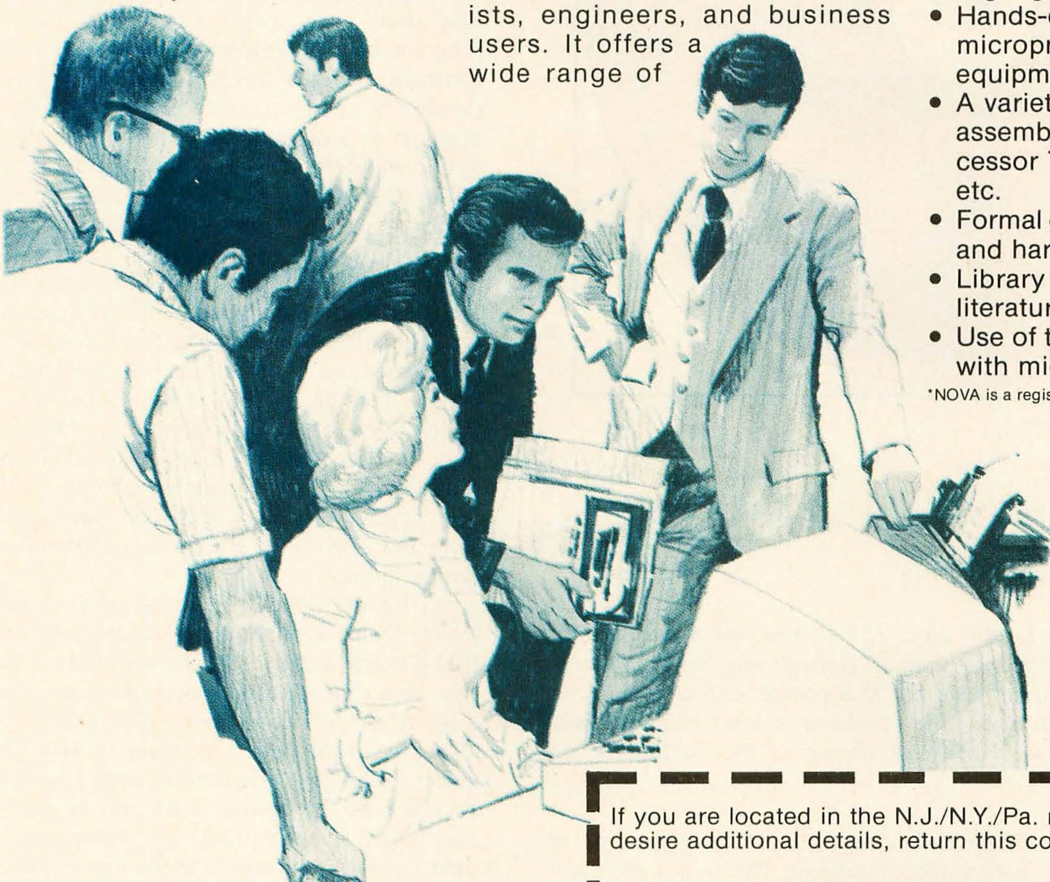
The new Typetronic Computer Store has all the professionalism and depth of the field-proven Typetronic minicomputer management systems. In fact, the store is staffed by experienced professionals who have devoted years to the success of the Typetronic systems... powerful tools for management in such fields as retail chain stores and chemical distributor operations.

The Typetronic Computer Store is New Jersey's most complete and comprehensive store for hobbyists, engineers, and business users. It offers a wide range of hardware and software packages, including microprocessor and NOVA* compatible minicomputer systems and peripherals... plus classroom courses, expert assistance, and many other services.

Highlights include:

- Hands-on demonstrations of microprocessors and other equipment.
- A variety of microprocessor assembly kits... IMSAI, Processor Technology, Motorola, etc.
- Formal classes on both software and hardware topics.
- Library of publications, product literature, etc.
- Use of the Typetronic computer with micro-assembler.

*NOVA is a registered trademark of Data General Corp.



typetronic computer store

806 Route 17, Ramsey, N.J. 07446
201-825-1300

If you are located in the N.J./N.Y./Pa. metropolitan area and desire additional details, return this coupon:

( ) We are considering the automation of our company operations. Please have a systems representative contact me.

( ) Please send me your brochure on the new Typetronic Computer Store.

Name ______________________ Title ____________

Company ______________________ Div. ____________

Street ____________________________________

City ______________ State __________ Zip ________

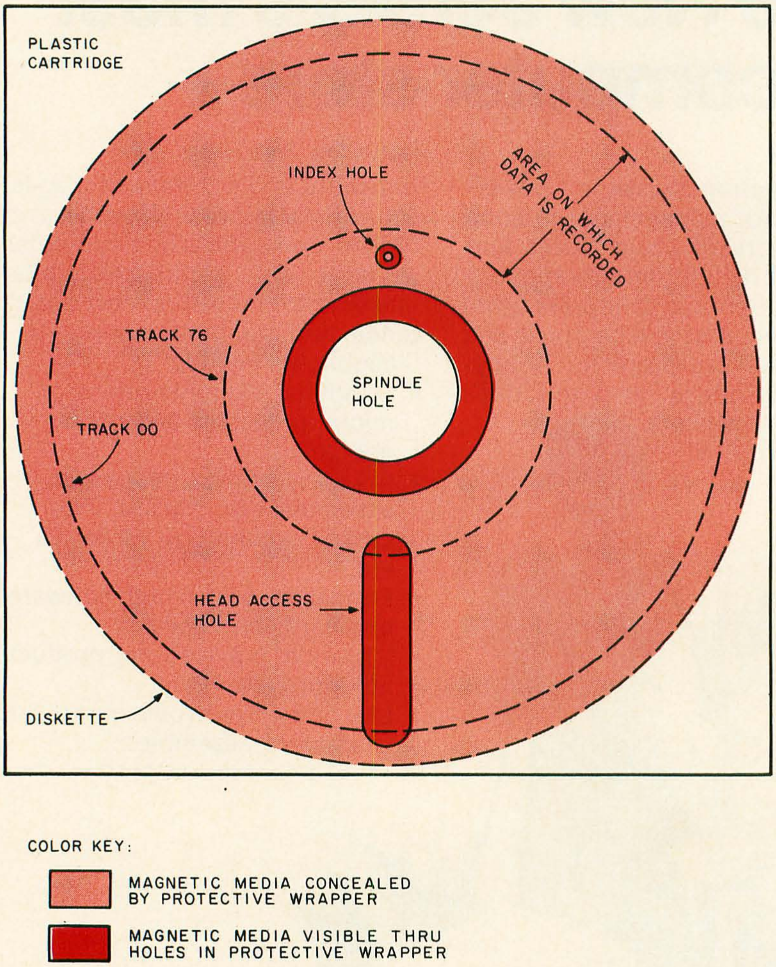


Figure 1: An X-ray view of a floppy disk. The floppy disk is encased in a protective plastic cartridge 8 inches square. Dotted line circles illustrate the portion of the disk where data is stored. It is coated with an oxide layer similar to that used on magnetic tape. The combination read and write head contacts the oxide layer through the slot shown at the bottom of the cartridge.

disks and cassette systems in terms of performance and cost. (We're not even considering the much more expensive 2311, 2314 and 3330 type disk drives of the big machines here.) Both floppies and digital cassette systems provide high performance at medium cost. Audio cassettes provide low performance at low cost. (Cartridge disks are not likely to fall into the hands of a hacker; they are included for reference only.) The big tradeoff is speed versus price, of course. A floppy is very fast and more expensive, while the digital cassette is much slower but cheaper. The same holds true for the media. Diskettes cost more per kilobyte than do high quality audio cassettes like Memorex MRX-2 C30, when used for digital recording. If speed and random access are important, get a floppy. If low cost and vast amounts of storage are important, get digital cassettes. My own system will eventually have a floppy disk for an operating system and a digital cassette transport for archival storage.

Floppy disks were first developed at IBM laboratories in the middle 60s. They were first used as a means of storing microprogram code for programmable peripheral controllers in the S/370 family, and were eventually used to store the microcoded diagnostics and emulator functions for the 370 computers. Then, in the beginning of 1973, IBM introduced the 3740 Data Entry System. Like a keypunch, except that data is recorded on a floppy instead of cards, one 3740 floppy disk holds the equivalent of 3000 cards. IBM then predicted that floppies would replace cards as the principle data entry medium. This of course has yet to happen completely, but is well under way. In the midst of all the commotion, someone discovered that floppy disks make excellent mini and microcomputer peripherals, and the floppy peripheral industry was born. Thus the 3740 is the device that started the floppy industry.

As for the floppy disks (diskettes) themselves, figure 1 provides a closeup view. An IBM diskette looks for all the world like a 45 rpm record in a plastic jacket. Actually, the diskette is defined to be a disk of heavy Mylar based magnetic tape material. It is 7.8 inches (19.8 cm) in diameter with a center spindle hole 1.5 inches (3.81 cm) in diameter. The 0.01 inch (0.025 cm) index hole is used to synchronize data as the disk rotates. Data is stored on the surface of the disk in the oxide coating, the same technique used in magnetic tape storage. And like magnetic tape, floppies are very susceptible to contamination by foreign particles (dirt, dust, fingerprints). Therefore, the diskette is enclosed in a thin semistiff low friction plastic jacket known as a cartridge. The standard cartridge has three openings in it to allow the spindle, read and write head, and index photosensor to have access to the disk.

The inside surfaces of the cartridge have soft, low friction liners. The liners wipe the



UP AND RUNNING

TDL EQUIPMENT USED BY NEW JERSEY PUBLIC TELEVISION
TO PROCESS NEW JERSEY GUBERNATORIAL PRIMARY ELECTION RETURNS

John Montagna, computer engineer (above left), lead this successful network team in generating election results speedily, efficiently and reliably using predominantly TDL hardware and software. Montagna created three programs to get the job done. The text for a SWAPPER program was written and assembled using the TDL TEXT EDITOR and Z80 RELOCATING MACRO ASSEMBLER. The SWAPPER text and all debugging was run through TDL's ZAPPLE MONITOR. The relocatable object code was punched onto paper tape. A MAIN USERS program updated votes and controlled air display. An ALTERNATE USERS program got hard copy out and votes in. The latter two programs were written in BASIC. Montagna modified the ZAPPLE BASIC to permit timesharing between the two USERS programs.

Four screens were incorporated, two terminals entered votes as they came in and were used to call back votes to check accuracy. Montagna called on the power and flexibility offered by TDL's ZPU board and three Z-16 Memory boards.

Montagna's setup worked constantly for over four hours updating and displaying state-wide and county-wide results without flaw.

"I chose TDL because they have all the software to support their hardware, and it's good; it has the flexibility to do the job."
John Montagna

We salute John Montagna and NEW JERSEY PUBLIC BROADCASTING for spearheading the micro-computer revolution.

TDL's XITAN SYSTEMS have the capacity to do similar tasks for you. Write to us for XITAN information and the name of your nearest TDL dealer.

TECHNICAL DESIGN LABS

RESEARCH PARK BLDG. H 1101 STATE ROAD
PRINCETON, NEW JERSEY 08540 (609) 921-0321

Circle 117 on inquiry card.

Figure 2a: A diagram of a floppy disk showing a typical location of the wire protect (or file protect) hole. When this hole is punched out, data cannot be erased from the disk. The write protect hole serves the same purpose as the plastic tabs on a magnetic tape cassette.

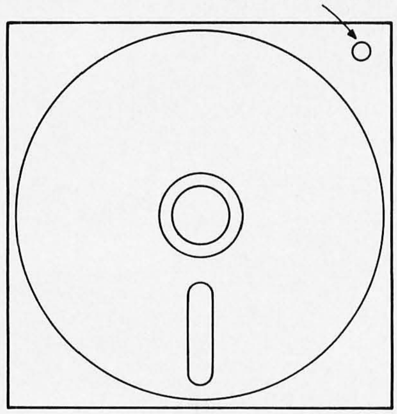
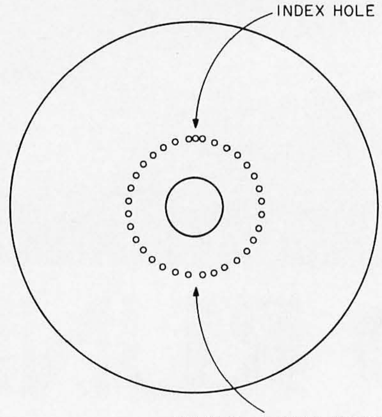


Figure 2b: A floppy disk with a design modification to allow "hard sectoring." 32 0.10 inch (0.025 cm) holes are punched in the disk in addition to the index hole. These holes enable the hardware to detect the exact rotational position of the disk by means of a photosensor. The need for space wasting address fields is eliminated with one hole per sector, and the number of data sectors can be typically increased from 26 to 32 per track.



disk clean and eliminate static charge while reducing the necessary spindle torque. To protect the disk even further when the diskette is not in use, it is kept in a cardboard sleeve, in the same manner that an LP record is kept in its album cover. The industry standard (IBM 3740) specifications for diskettes is fairly rigorous and widely accepted, although a number of variations have lately appeared on the scene. Two of these are shown in figure 2.

The write protect hole is a small plastic knockout in the cartridge which serves the same purpose as the small knockout tabs in a tape cassette. A sensor detects whether the knockouts have been removed or not, and if so, disables the write electronics. Another variation is the presence of many small sector indexing holes punched in the disk. The advantage of this so-called hard sectoring process is a higher data capacity per track. This will be discussed in more detail later.

A third, relatively recent, innovation in the manufacture of diskettes is to coat both sides of the Mylar backing with magnetic oxide and to put an additional head access slot on the opposite side of the cartridge, giving a two-sided floppy disk (see photo 2 and figure 6). Some manufacturers have gone so far as to develop a drive mechanism which simultaneously accesses both sides at once. IBM, Information Terminals, BASF and Wabash are some of the companies which sell floppy disks. The price of diskettes varies from \$12 to \$7 each in boxes of ten to about \$4 to \$5 each in large quantities.

Drive Hardware

There is a commonly drawn analogy between a disk drive and a phonograph. Both are mass storage devices in which the data is stored on a platter-shaped medium. The platter is seated on a spindle around which it revolves while the data pickup (needle or magnetic head) moves radially across the data. In the case of a phonograph record, the music, or data if you will, is recorded on spiral grooves or tracks which are cut into the record. To access a particular song, the tone arm is moved radially over the record and placed down in the starting groove of the song. In contrast, the data on a computer disk is stored in discrete concentric circles, not one continuous spiral. The concentric circles are called tracks and are accessed by a magnetic head which is bumped mechanically from track to track under computer control. Data is stored by means of saturated magnetic recording at a maximum density of 3200 bits per inch (on the innermost track), and 48 tracks per radial inch.

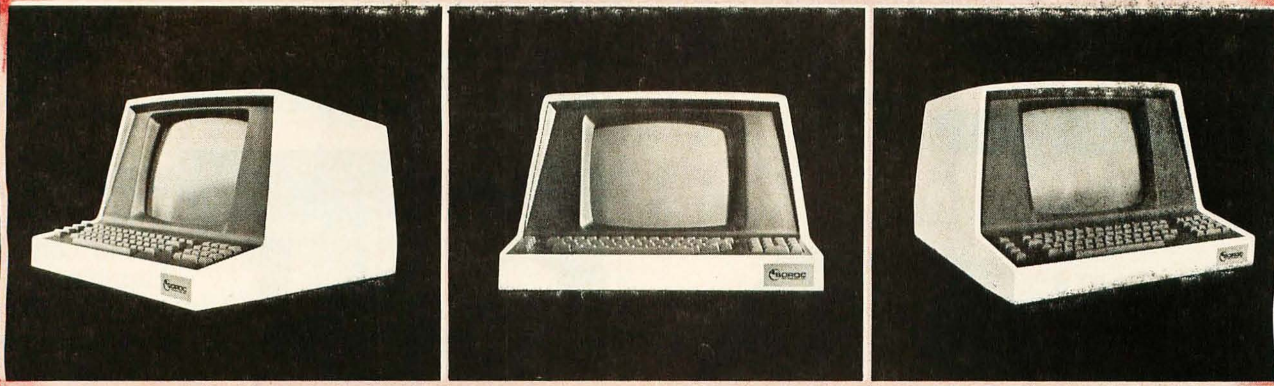
Most floppy drives on the market today use either a synchronous AC or servo-controlled DC motor to drive the spindle at exactly 360 rpm. Speed control is very important to insure data reliability. The spindle motor runs continuously whether or not a diskette is loaded.

When a diskette is loaded into a drive, it is inserted into a narrow slot and the cartridge is held in place between a spring and a small metal protrusion. The door over the slot is then closed, clamping the diskette to the rotating spindle. At the same time, in most drives, the read and write head is engaged.

The type of read and write head used is one of the important differences between hard disks and floppies. Hard disks use a system known as flying heads in which the read and write heads are aerodynamically floated off the surface of the disk. They are held a very small and precisely controlled

Continued on page 35

WANTED



IQ 120

DESCRIPTION

- Lower Case
- Tabbing
- Dual Intensity
- 24 Line x 80 Char.
- Numeric Pad
- Auto Repeat
- Aux. Port
- Optional Printer Port & Block Mode
- Protect Mode
- 15 Baud Rates
- Addr. Cursor

at $995.00 the

SOROC IQ 120 is the most wanted terminal on the market.

For information concerning the REWARD contact...

SOROC TECHNOLOGY, INC. 165 FREEDOM AVE., ANAHEIM, CA 92801
714-992-2860 / 800-854-0147

Circle 108 on inquiry card.

Ohio Scientific advances the state-of-the-art of small computers.

From our inexpensive 8K BASIC in ROM Challenger IIP to our powerful triple processor Challenger III, Ohio Scientific offers a full range of products that are technologically superior to anything available on the market today.

Challenger IIP

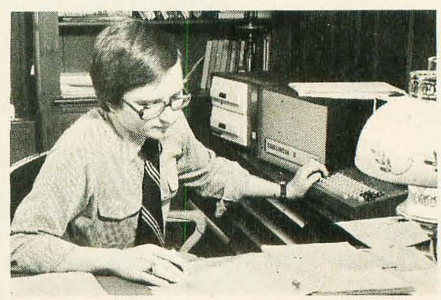


Challenger IIP from Ohio Scientific is our unique personal computer with BASIC in ROM and 4K RAM for programs in BASIC.

Complete with audio cassette interface and a full computer keyboard, Challenger IIP can be connected to a home TV via an RF converter and it's ready to go.

Challenger IIP comes fully assembled and tested for only **$598.00.**

Challenger II



Challenger II from Ohio Scientific is a disk based computer capable of storing up to 500,000 bytes of information on an Ohio Scientific dual drive floppy disk.

Challenger II comes with 16K of RAM (the disk BASIC is automatically loaded into the computer so there is no need for ROM's) and our powerful Disk Operating System (DOS) which allows the computer to perform big computer functions like random access, sequential and index sequential files in BASIC, and I/O distributors which support multiple terminals and industry standard line printers.

And best of all a 16K Challenger II with serial interface, single drive floppy disk, (250,000 bytes) BASIC and DOS costs only **$1,964.00** fully assembled.

Challenger III



Challenger III from Ohio Scientific is the revolutionary, new triple processor computer that allows you to run programs written for the 6502A, 6800 and Z-80 processors.

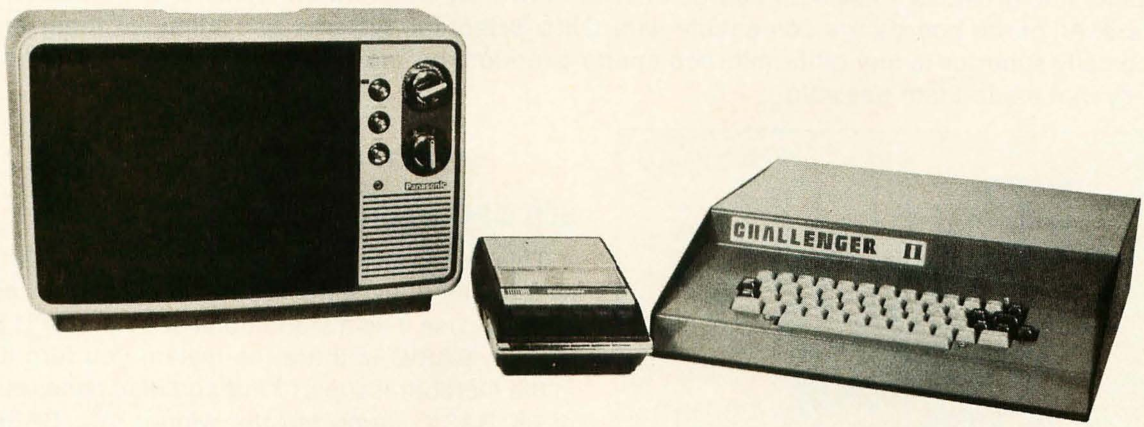
Incredible as this is, a disk based Challenger III costs only about 10% more than conventional single processor microcomputers. A 32K Challenger III with a serial interface and a dual drive floppy disk assembled and tested costs **$3,481.00.**

OHIO SCIENTIFIC

11679 Hayden • Hiram, Ohio 44234

For more information send for our Free, short form catalog, or send $1 for our 64 pg. Small Computing Buyers Guide.

Meet Challenger IIP from Ohio Scientific.



Unlike any other personal computer available today

Complete with BASIC in ROM and 4K RAM, Challenger IIP is the ideal computer for programs in BASIC.

BASIC is there the instant you turn the computer on with a full 32 x 64 character video display. Challenger IIP also comes with an Audio Cassette Interface for program storage. The user simply connects a Video Monitor or a TV via an RF Converter (not supplied) and the machine is ready to use.

Challenger IIP is ideal for both the home user who is new to computing or the experienced user who wants expansion capabilities. Challenger IIP comes with a four slot backplane and is expandable via the full Ohio Scientific product line, which includes 15 system boards offered in over 40 different versions.

Ohio Scientific has always maintained upward expandability from old models to new models, which is nice to know considering the rate at which technology is constantly improving. For example, Ohio Scientific's original 400 series products can be plugged right into the new Challenger IIP. And Ohio Scientific has 2 years of experience in building personal computers, so we're not new to this business unlike some of our competitors.

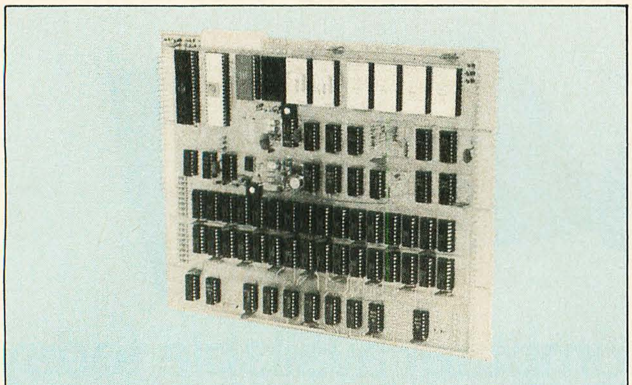
Complete with a full computer keyboard Challenger IIP comes fully assembled for $598 from Ohio Scientific.

Check the chart below and compare Challenger IIP with other BASIC in ROM computers. Unlike other personal computers, Challenger IIP has a much greater capacity for expansion and the capability to perform big computer functions with all of its big computer features.

	Ohio Scientific Challenger IIP	Other BASIC in ROM Computers
Processor	6502A	6502 or Z-80
Clock	1 or 2 MHz	slower
Display (Lines/Characters)	32/64	25/40 or 16/64
Keyboard	Full Computer (Capacitive Contact)	4 Function Calculator Type or Full Computer (Mechanical Contact)
Display Characters	256	128 or 64
Lower Case	Yes	No
Plotting	Yes	Yes
Audio Cassette Interface	Yes	Yes
BASIC	8K By Microsoft	some have only 4K BASIC
String Functions PEEK, POKE, User	Yes	Not Always
Machine Language Accessible	Yes	Not Always
Optional Assembler/Editor	Yes	No
Disk Option Available Now	Yes	No
In Case Memory Expansion Ability	36K	Less
Expansion Boards Available Now	15	None

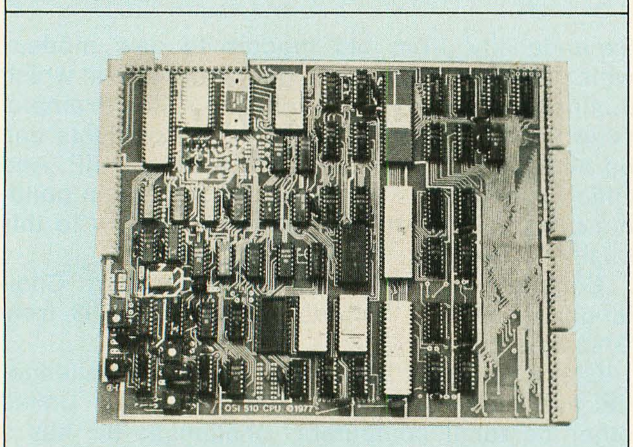
Introducing three boards only Ohio Scientific could build.

Ohio Scientific provides 15 system boards offered in over 40 different versions for Ohio Scientific Computer users. All of the boards are compatible with Ohio Scientific systems and many of them are by far technologically superior to any other microcomputer products on the market. And Ohio Scientific has the technology that made them possible.



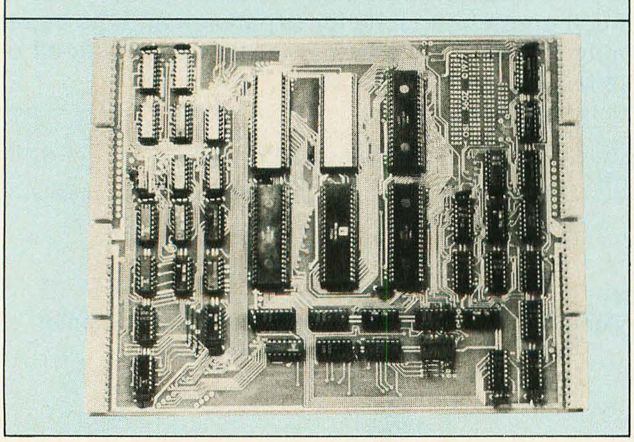
500 CPU Board

This board gives you our ultra-fast 8K BASIC in ROM with plenty of user workspace (4K RAM) for as little as $298.00. Use it as a standalone or as the CPU in a large system. BASIC is there the instant you turn it on. And in the October issue of *Kilobaud Magazine,* our version of 8K BASIC came out the winner in a BASIC timing comparison test of all of our competitors. The 500 is the fastest around!



510 Systems CPU Board

This is our unbelievable triple processor board! Complete with the 6502A, 6800, and Z-80 processors, this board allows you to run virtually all programs published for small computers. Available in the Challenger III, the 510 board is ideal for industrial development and research applications. There isn't another triple processor board like the 510 anywhere, except at Ohio Scientific!



560Z CPU Expander Board

The 560Z board is our multiprocessing board with a Z-80 and 6100 chip. This board allows you to run several processors simultaneously and the 6100 chip lets you run powerful PDP8 software with the 560Z. The 560Z board is the only multiprocessing board available for small computers, and Ohio Scientific makes it!

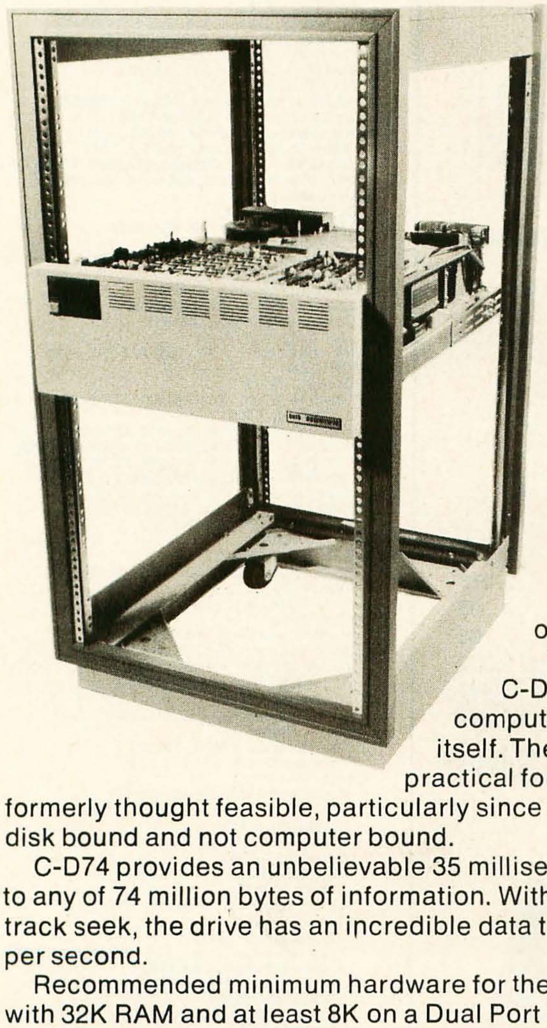
These three state-of-the-art CPUs are only a small part of the picture. Ohio Scientific's advanced technology offers you other unique features such as Multiport Memories, Distributed Processing, Big Disks with up to 300 megabytes on line, and Advanced Software.

OHIO SCIENTIFIC

11679 Hayden • Hiram, Ohio 44234

Announcing the most advanced disk anywhere for $6,000

The 74 megabyte disk from Ohio Scientific



C-D74 from Ohio Scientific is the ultimate storage device for small computers.

The C-D74 is the first Winchester technology disk for small computers making big system technology affordable and reliable for the small system not under maintenance contract.

The disk uses a non-removable sealed chamber drive with a unique rotary positioner to provide the highest performance disk available today.

The Ohio Scientific C-D74 can store all the records of a medium size company for instant access. And the Winchester technology of the C-D74 means that the drive can run 24 hours a day without worry of disk wear.

There are other important C-D74 applications in business computing and research in computing itself. The disk makes small computers practical for much larger jobs than formerly thought feasible, particularly since most business computing is disk bound and not computer bound.

C-D74 provides an unbelievable 35 millisecond average access time to any of 74 million bytes of information. With a 10 millisecond single track seek, the drive has an incredible data transfer rate of 7.3 megabits per second.

Recommended minimum hardware for the C-D74 is a Challenger with 32K RAM and at least 8K on a Dual Port 525 board, and a single or dual-drive floppy disk.

The drive, cable, interface for an Ohio Scientific Challenger and OS-74 operating system software is $6,000 FOB Hiram, OH. Equipment rack shown not included.

OHIO SCIENTIFIC DEALERS

Abacuz Stores Limited
55 Erb St. East
Waterloo, Ontario
Canada N2C 3E0
(519) 885-1211

American Microprocessors Equipment & Supply Corp.
20 N. Milwaukee Ave.
Prarieview, IL 60069
(312) 634-0076

Century 23
4566 Spring Mountain Rd.
Las Vegas, NV 89102

Computer Mart of New York
118 Madison Ave.
New York, NY 10010
(212) 686-7923

Computer Power
P.O. Box 28193
San Diego, CA 92128
(712) 746-0064

Delaware Microsystems
92 E. Main St. #1
Newark, DE 19711
(302) 738-3700

Desert Data
Microcomputer Sales
P.O. Box 1334
Tucson, AZ 85702
(602) 623-6502

The Home Computer Co.
P.O. Box 1891
University Station
Charlottesville, VA 22903
(804) 295-1975

Microcomp
P.O. Box 1221
Fond-Du-Lac, WI 54935
(414) 921-4669

Microcomputer Workshop
234 Tennyson Terr.
Williamsville, NY 14221
(716) 634-6844

Omaha Computer Store
4540 S. 84th St.
Omaha, NE 68127
(402) 592-3590

REPRESENTATIVES

Abacuz Data
P.O. Box 276
Oil City, PA 16301

Associates Consultants
33 Ogden Ave.
East Williston, NY 11596
(516) 746-1079

BRAG Microcomputers
19 Cambridge St.
Rochester, NY 14607
(716) 442-5861

Computer Business
P.O. Box 171
LaPorte, IN 46350
(219) 362-5812

Johnson Computer
P.O. Box 523
Medina, OH 44256
(216) 725-4560

Omega Computing, Ltd.
Box 220, Station P
Toronto, Ont. M5S 2S7
(416) 424-2174

Pan Atlantic Computer Systems, GmbH
61 Darmstadt
Frankfurterstrasse 78
West Germany
(08102) 3206

Science Education Extension
11516 LeHavre Dr.
Potomac, MD 20854
(301) 299-9506

Spectrum Technology Services
P.O. Box 942
Palos Verdes Estates, CA 90274

Tek-Aids, Inc.
1513 Crain St.
Evanston, IL 60202
(312) 328-0110

Secom Systems
541-1 New Peach Tree Rd.
Chamblee, GA 30341
(404) 934-3272

Yingco, Inc.
2 World Trade Cntr.
Penthouse 107th Floor
New York, NY 10048
(212) 775-1184

The state of the art in small computers.

To order direct call 1-216-569-3241

Continued from page 28

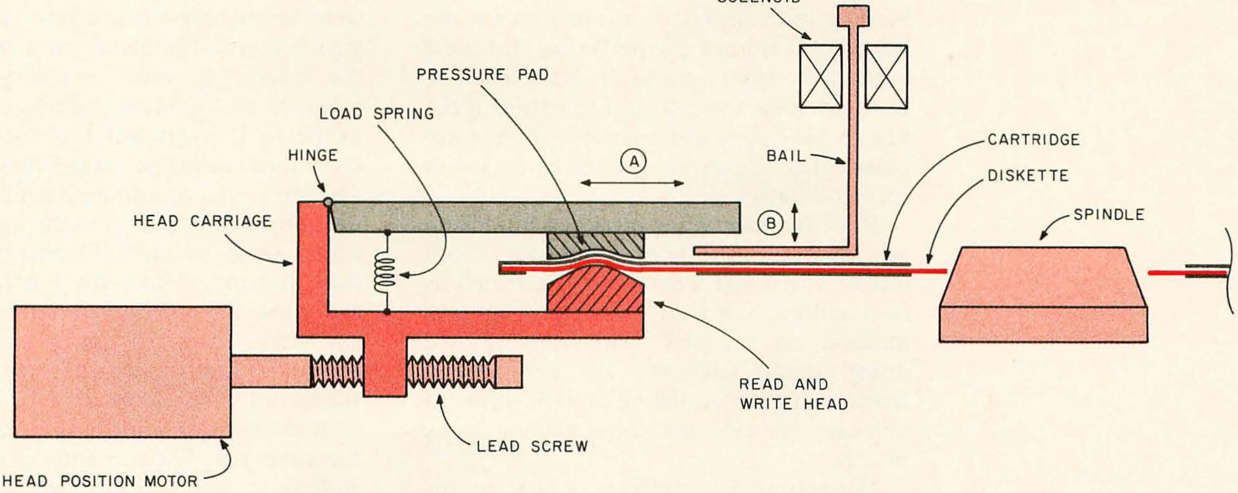


Figure 3: Diagrammatic representation of a floppy disk drive system. The read and write head and the pressure pad are mounted on the head carriage, which can be moved radially (see arrow A) to any given data track by means of the motor controlled lead screw. The bail is solenoid actuated (see arrow B) and is used to lift the pressure pad away from the disk during insertion and removal. The floppy disk remains in its protective plastic cartridge at all times; the head makes contact with the oxide surface through an access slot (see figure 1). The spindle rotates the floppy disk at 360 rpm during use.

distance away from the magnetic surface in the relative wind created by the rapidly spinning disk. Flying heads are very difficult to build and maintain, and therefore quite expensive. The advantage of flying heads is that they cause no wear on the disk surface (unless they "crash," of course) and therefore permit very high rotational speeds and data rate.

Floppies, on the other hand, use a contact head. Contact heads are much simpler, mechanically speaking. They are pressed onto the floppy in much the same way as a tape head is pressed into magnetic tape. Thus in the terminology of flying heads, floppy disk heads are permanently "crashed." The chief drawback of these contact heads is the wear they cause to the floppy and vice versa. Diskettes are rated in terms of the number of passes the head makes over a particular spot before an error is likely to occur there, while read and write heads are rated in terms of the number of hours they can survive contact with the diskette before replacement. Typically quoted values of component life are in the millions of passes per track for diskettes, and tens of thousands of hours of contact for the read and write heads. In order to stretch the useful component lives, most floppy controllers will command the drive to unload the head from the diskette when the floppy is not being used. A typical mechanism for head loading and positioning is shown schematically in figure 3.

Most manufacturers use a lead screw driven by a stepping motor to move the head from track to track (A stepper motor is a motor which rotates a fixed number of degrees every time it receives a pulse.). The head is mounted on a carriage which is pushed back and forth across the diskette by the rotation of the lead screw. The pitch of the lead screw is chosen such that the stepper motor's angular rotation is translated into a linear motion equal to the track-to-track radial distance of 0.0213 inches (0.05 cm) (IBM format). A "seek," which means moving the head to the desired track, consists of the retraction of the head back to track 00 followed by n pulses to the stepper motor in order to reach the desired track n. Track 00 position of the head is detected by a microswitch sensor. Likewise, tracks 44-76 are detected, for reasons to be discussed later. Variations on head positioning schemes in the newer drives include "voice coil" methods which position the head with a linear actuator similar to an acoustic suspension loud speaker's voice coil.

Controllers

The basic function of a floppy disk controller is to do many of the small house-

keeping tasks necessary in order to use the floppy as a storage device. Perhaps the most important task of a controller is the handling of floppy disk formatting. Formatting is the key to the floppy disk's random access capability, since it provides the stored data with access addresses.

Basically, formatting breaks each track up into discrete areas known as sectors. Each sector is of fixed length and is assigned its own address based on the sector's physical location on the disk. With soft sectored disks, sector addresses are permanently written into the beginning of each sector to uniquely identify the corresponding block of data.

Formatting is a concept similar to the wide bands between selections on a phonograph record. The bands on a record allow the listener to select and play particular songs. In fact, a recent addition to the list of exotic hi fi equipment is a microprocessor controlled turntable called the Accutrac. The unit has an infrared emitter and reflection detector built into the tonearm head which sense when the tonearm is over a smooth band, detect the record's format, and allow the processor to randomly access any track. The track can be played and replayed, or any other track on that side of the record can be accessed.

A floppy disk controller works in much the same way. The computer passes a sector address to the controller, which moves the

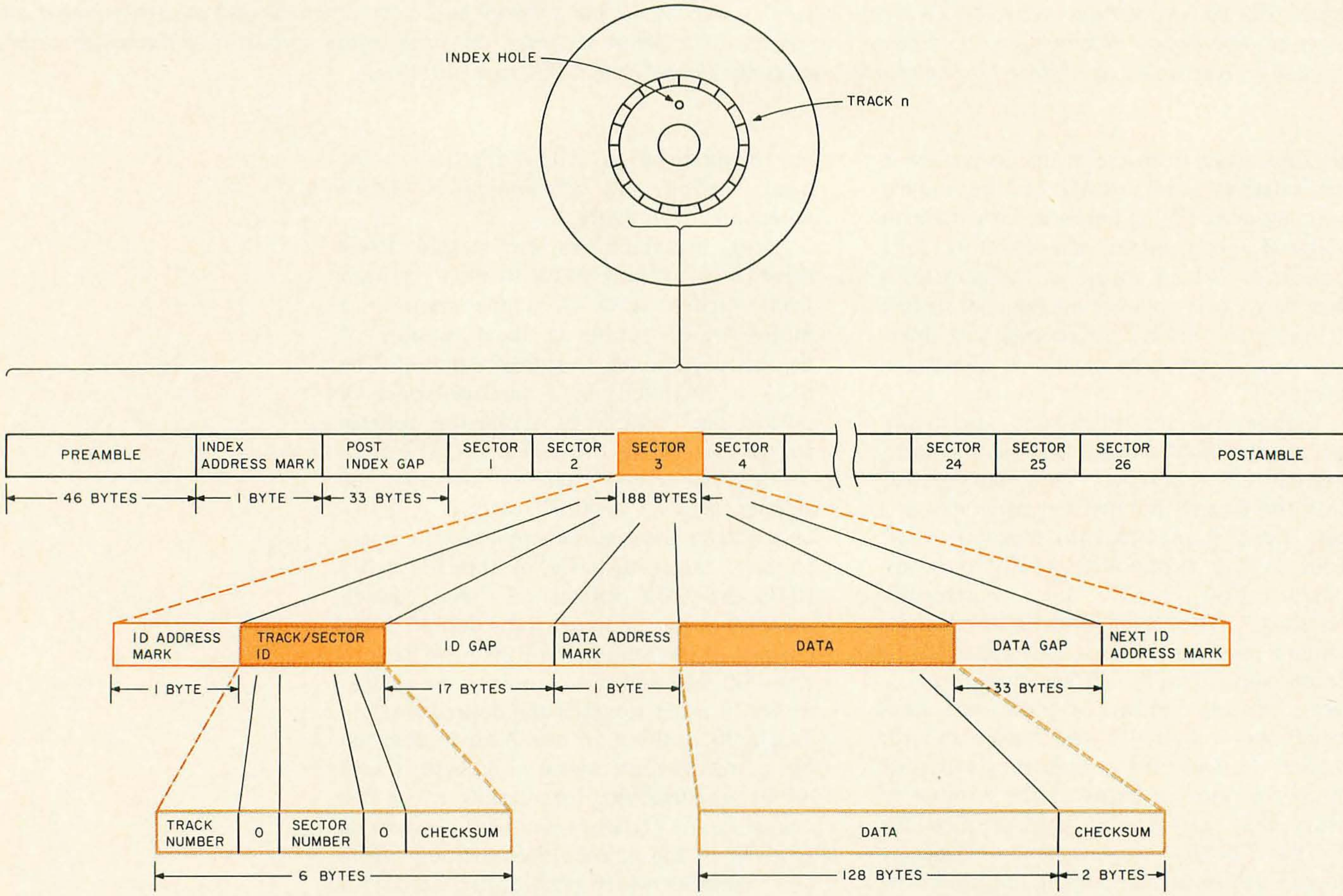
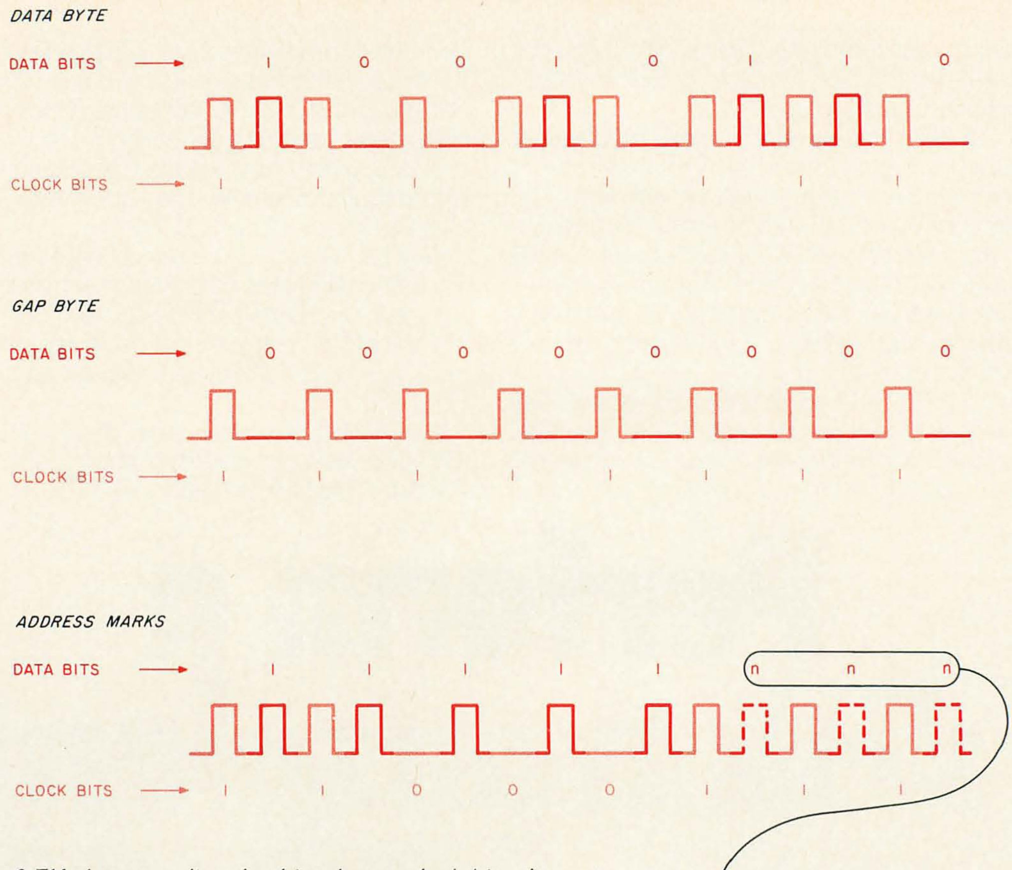


Figure 4: The IBM 3740 floppy disk format. The format is "soft" sectored, meaning that data written in the track controls the organization of information on each track. Each of the 77 data tracks on the floppy disk contains data, address and control fields grouped together to form "sectors." Each sector contains a sequence of fields, identical to those of the other sectors, which are further broken down into individual data bytes. One complete track is shown in this illustration. The index hole provides the only hardware synchronization in this format.

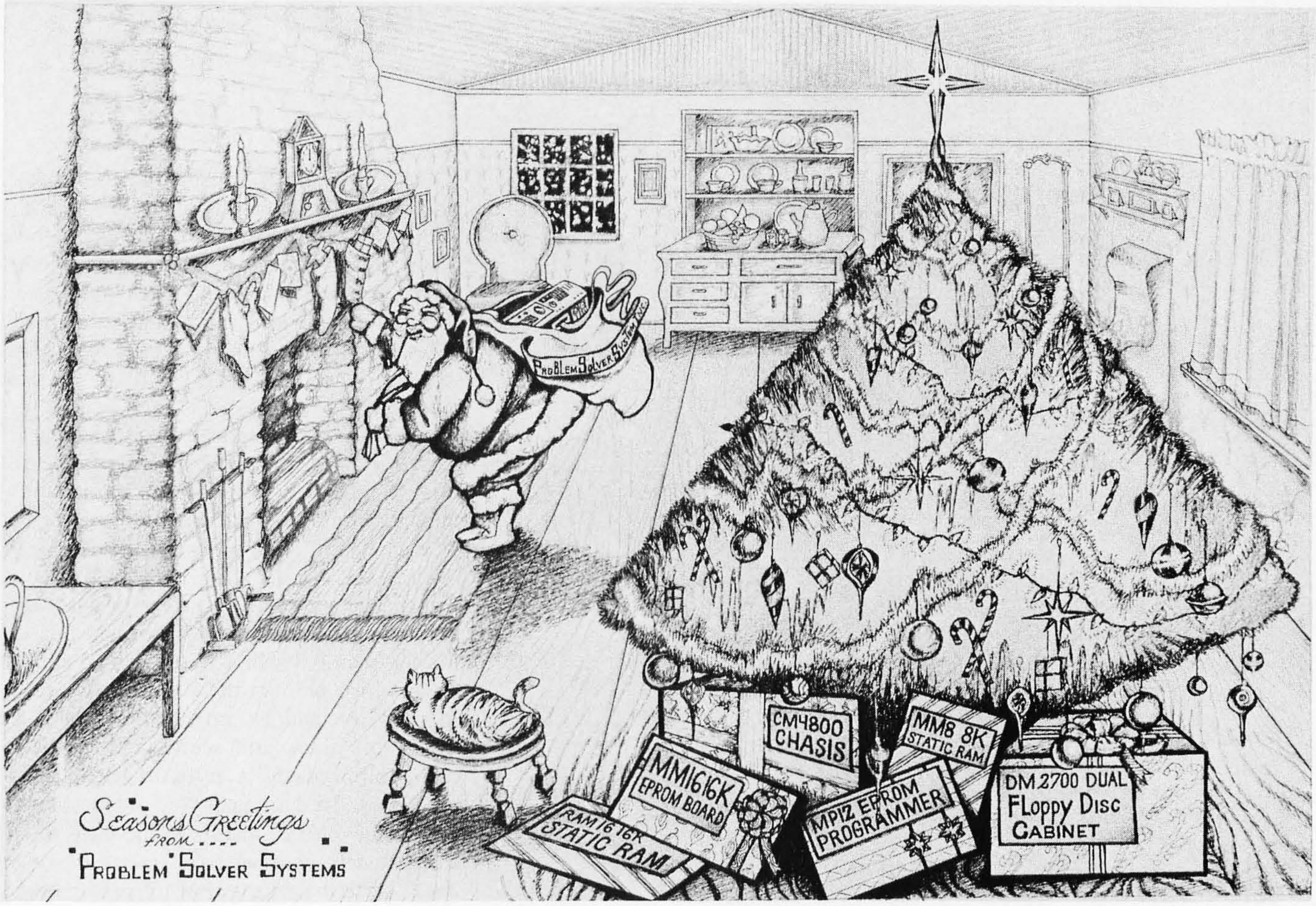


n n n	MARK TYPE
0 0 0	DELETED DATA
0 1 1	DATA
1 0 0	INDEX
1 1 0	ID

Figure 5: Examples of FM data encoding. In this scheme, clock bits alternate with data bits to provide constant resynchronization during data recovery. The concept of FM, or "frequency modulation," comes about because a string of 1s in the data bit, when interlaced with the clock bits, gives a different frequency of pulses than a string of 0s. The pulse train at the top of the figure shows the data byte 10010110 encoded using this technique. The middle section shows the format for a "gap" or zero byte. Gaps are used to provide buffer regions between fields so that minor fluctuations in motor speed will not affect accuracy. Also shown is the format for address marks, which serve to inform the system that the next byte is the beginning of a data or address field. Certain clock and data bits of the address marks are intentionally set equal to zero in order to differentiate them from other types of bytes. The last three data bits in the address mark tell whether the information that follows is deleted data, regular data, an index byte or an ID byte.

head to the proper track. The controller then waits for the desired sector address to pass under the head before allowing the data transfer to occur. The data transfers are always of fixed length to avoid unintentional overwriting of other sectors.

The IBM standard format is illustrated in figure 4. It currently represents the most popular form of soft sectoring. The format is conservatively designed to protect against overwrites and to synchronize data transfers. The format details the content of each of the 77 tracks: data, address and control fields grouped together to form sectors. Each sector contains a sequence of fields identical to those of the other sectors. These fields are further broken down into individual data bytes, each of which is coded to identify its type to the controller. This encoding is easily accomplished through the use of clock bits. Clock bits are written, as shown in figure 5, prior to every data bit on a diskette. This is done to provide continuous resynchronization in the data recovery process. This in turn decreases the confusion resulting from small motor speed variations while decreasing the signal recovery bandwidth and improving the overall data reliability. Most data recovery circuits utilize a phase locked loop which provides a limited form of timing memory that can recover a data bit if the preceding clock bit is missing, or the motor speed is slightly off. Those bytes which serve a special formatting function are called address marks and are recognized by a pattern of intentionally missing clock bits. There is a single pattern of clock bits that serves to identify all four types of address marks; each type is iden-



Problem Solver Systems
CM4800 CHASIS
MM8 8K STATIC RAM
MM16 16K EPROM BOARD
RAM 16 16K STATIC RAM
MP12 EPROM PROGRAMMER
DM2700 DUAL FLoppy Disc CABINET
Seasons Greetings
from....
Problem Solver Systems

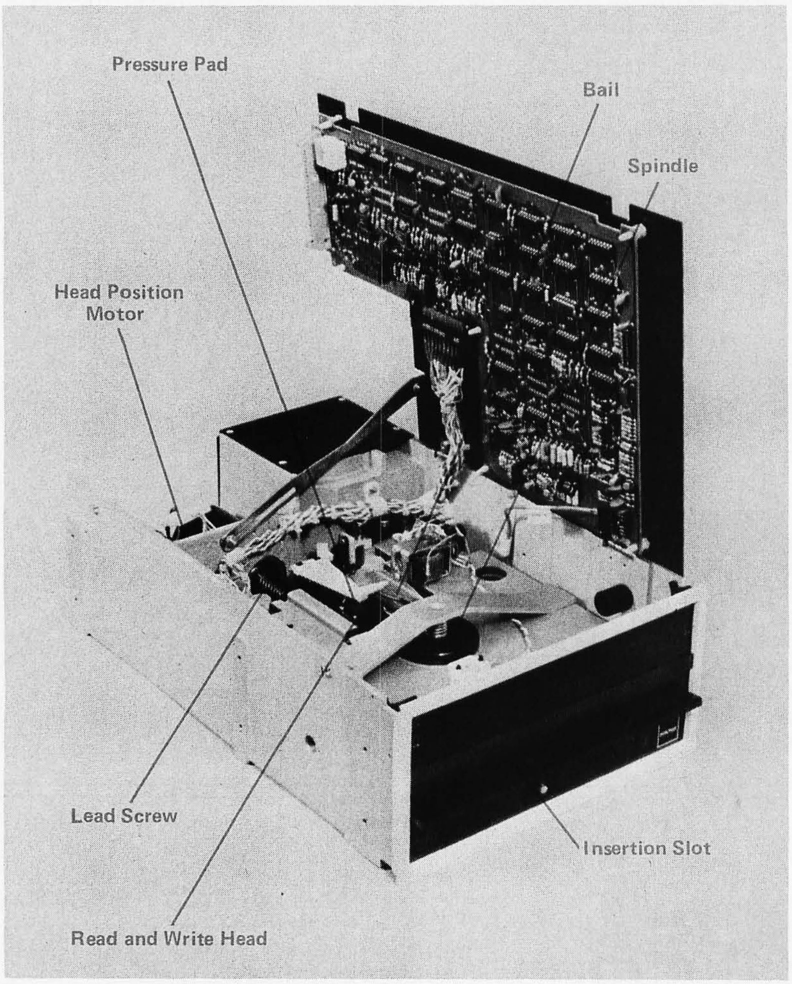


Photo 1: A floppy disk unit. The cover has been opened to reveal the disk drive system and its associated electronics. The floppy disk is inserted and removed at the front, as shown. Courtesy Innovex Corporation.

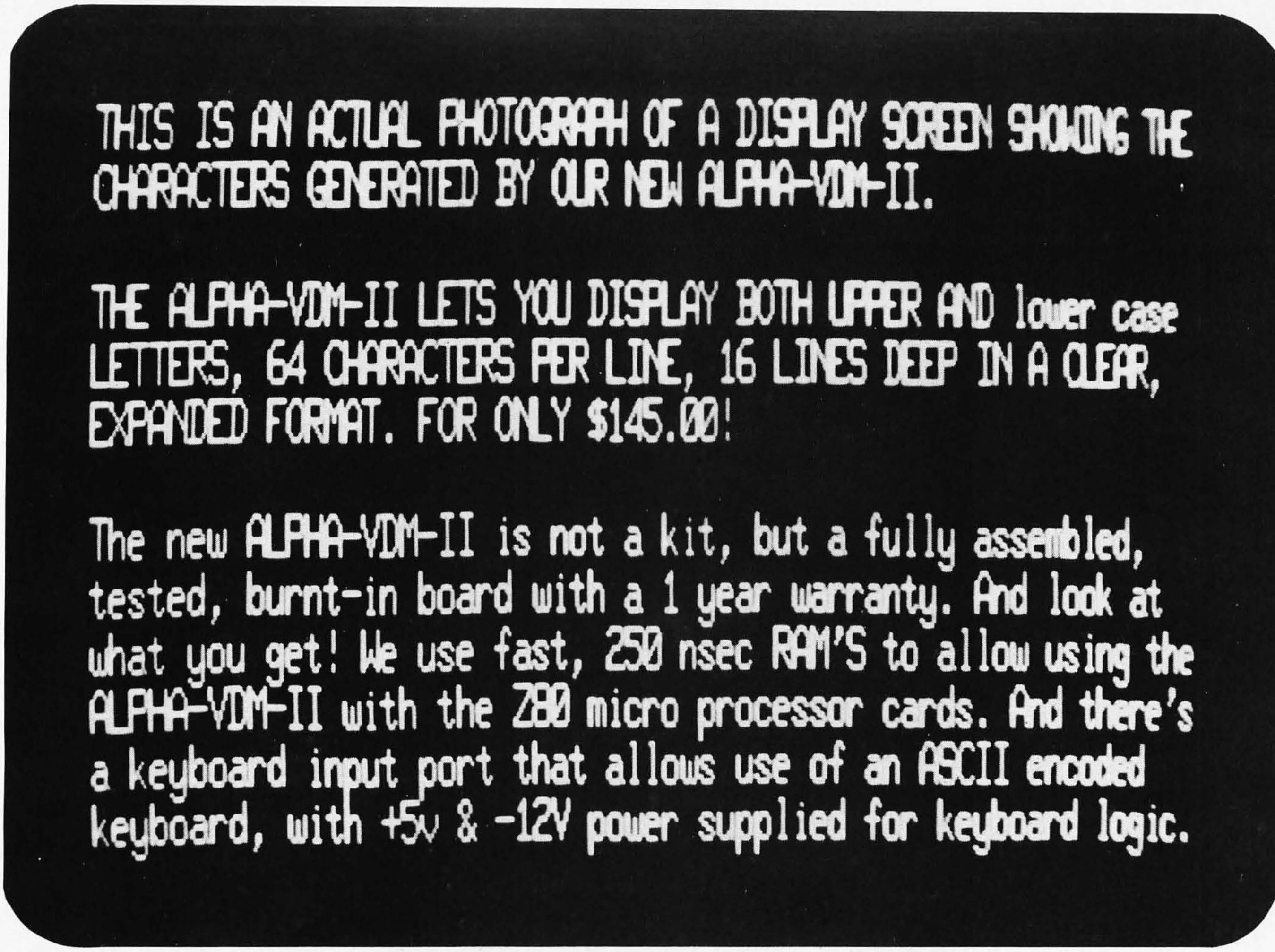
tified by the data bits in the byte. Address marks serve to inform the system that the next byte is the beginning of a data or address field.

Another special byte present in the IBM format is the zero byte, which makes up the predefined gaps. Gaps exist to provide a buffer region between other fields whose physical length may vary slightly, depending on the spindle speed and software synchronization. Gap bytes contain only clock bits which preserve the proper output frequency in the phase locked loop for synchronous data recovery, starting with the first byte of the next field.

Picking the index hole as a convenient starting reference, every track will contain the same sequence of data, address and control fields. Roughly 46 gap bytes after the leading edge of the index photocell there is an address mark called an index address mark. The index address mark byte serves as a landmark indicating that exactly 32 bytes follow before the first byte of the first sector of the track. The length of the post index gap is fixed at 32 bytes since it lies between the index address mark and the ID record of the first sector, and neither of these fields are user written. Consequently, they are immovable and fixed in length.

Each sector has four major fields: the ID record, the ID gap, the data field record, and the data gap. The ID record, as its name implies, provides the complete identification for the sector. Its first byte is an ID address mark which permits the system to get ready to read the address information. Next are the 8 bit track and sector address fields, each followed by a byte of eight zeros. Finally, a 16 bit cyclic redundancy check word (see BYTE, March 1977, page 42) is calculated to confirm error free address read back. Note that in the IBM format, the track and sector addresses are defined by the sector's physical location on the diskette. Because of this the user cannot usually modify the ID record. Following the ID record is the ID gap, which is 17 bytes long. Its primary functions are to buffer the length of the data field record and to provide a place for the read and write head to switch to write mode before entering the data area without disrupting any other information.

The data field record is the heart of the floppy disk, and without it none of the rest of these fields would be necessary. The data field consists of 130 eight byte bits in which the owner of the diskette is free to record programs, text, or any other information desired. Actually, only 128 bytes of the 130 bytes allocated is free for user data; the final two bytes of the data field are reserved for another cyclic redundancy check (CRC) word to permit error checking during read operations. After the cyclic redundancy check word, there is another gap, 33 bytes long, providing a head write-to-read transition area and another safeguard against data spillover. The next sector's ID address mark follows immediately after the data gap. Of course, the tangential speed of the diskette is different for every track, being greatest on track 00 and least on track 77. This physical fact, combined with a constant data rate of



The Alpha-VDM-II contains 1K (1024) bytes of random access memory, to which the processor can read or write, just as though the memory were an integral part of the system. As the information is written, the contents of this on-card memory are displayed instantly without interrupting the operation of the processor.

All timing required to generate a standard video signal is provided by a crystal oscillator and associated digital circuitry. Centering of the display on the monitor screen is controlled by drift-free counter logic.

The 1K by 8 static display memory buffer is directly addressable as RAM on the S-100 bus. Displaying data on the screen is accomplished by moving the data to be displayed in the first 512 bytes of the Alpha-VDM memory. Therefore the display update is essentially instantaneous. Output routines can make use of all Memory Reference instruction, including one byte moves. (i.e. MOV M, reg.)

Multiple programmable cursor circuitry is built in. All 1024 cursors can be displayed at one time, and anywhere in the display. Thus, the VDM can display white-on-black or black-on-white — perfect for many video games! The VDM also features EIA Video output for any standard video monitor, or a TV repair shop can easily modify your own set.

The VDM comes with free terminal mode software, designed for teletype replacement. Options include select blinking cursors, text line blanking after carriage return.

Also available: 4K RAM; $107.00, 8K RAM; $197.50, 8K(Z) fast RAM; $217.50, Alpha-VDM; $107.00, Graphics-VDM; $137.00.

Order direct, by check, BankAmericard or Master Charge (Add $1.50 shipping, credit customers give us all the card numbers, please and Ohio residents add 4½% sales tax) or contact us for more information. Kent-Moore Instrument Company, a subsidiary of Kent-Moore Corporation (founded in 1919), P.O. Box 507, Industrial Ave., Pioneer, Ohio 43554. (419) 737-2352. Or, Kent-Moore of Canada, 246 S. Cawthra Rd., Mississauga, Ontario L-A3P2, Canada.

Kent-Moore

INSTRUMENT COMPANY

Circle 68 on inquiry card.

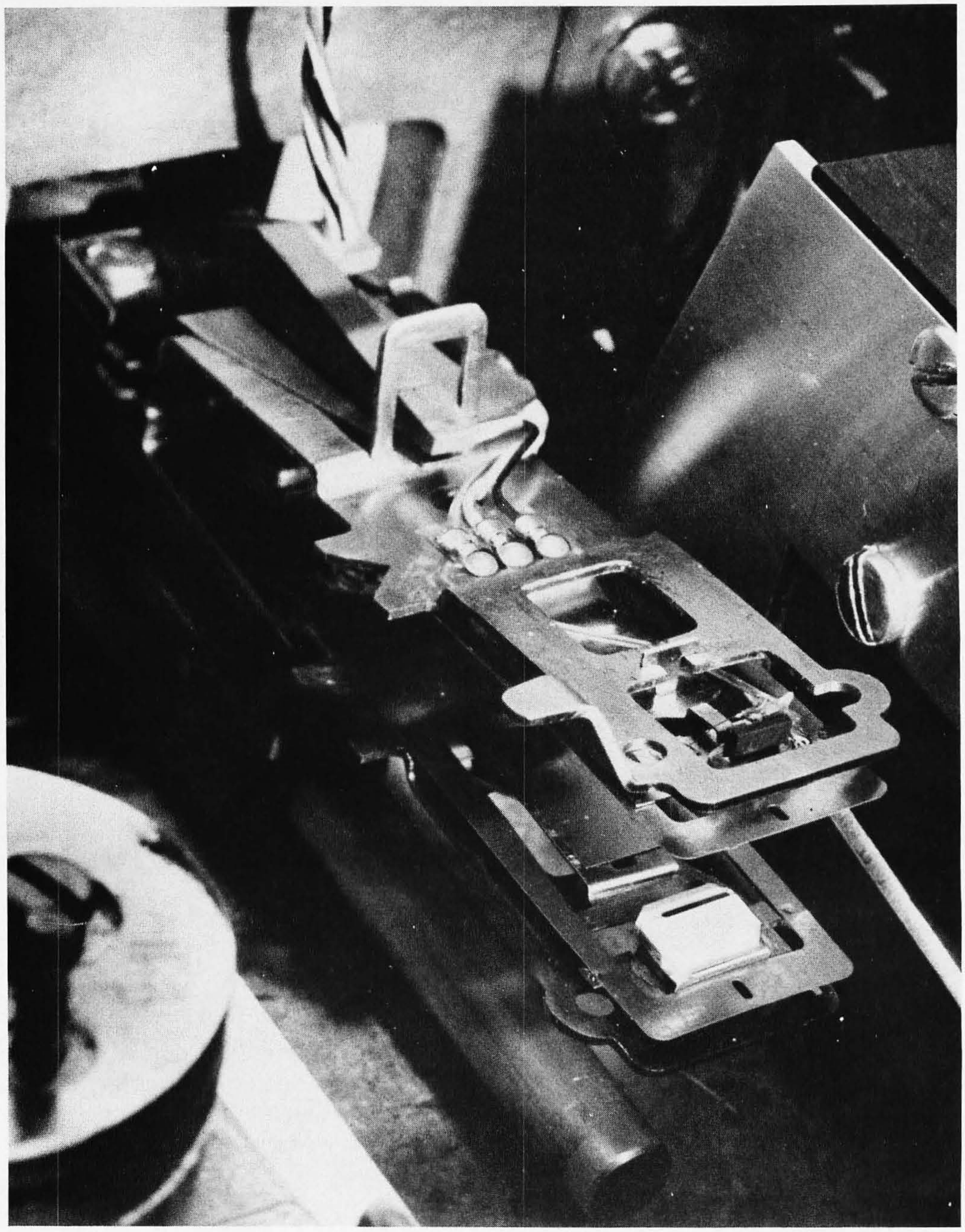


Photo 2: A closeup of the Shugart SA850 double-sided floppy disk head assembly. Each head in effect acts as the pressure pad for the other head. Courtesy Shugart Associates.

250 kHz, means that the physical bit density on each track will be different, track 77 having the highest density. In order to prevent peak shift distortion (see BYTE, February 1977, page 36) on the more crowded innermost tracks, the write current delivered to the head is reduced when writing on tracks 44-77.

Most floppies in use today use the frequency modulation (FM) encoding described here. In the search for ever higher performance, several new codes have been developed to increase the capacity and transfer rate of floppy disks. These schemes are MFM (modified FM) and M2FM (modified MFM) which are usually referred to as "double density" options by manufacturers. They work by a set of rules that remove clock bits when those bits will not be required for synchronization. Group coded data is also more compressed than raw FM. All of these techniques require more sophisticated electronics than FM, and are slightly less reliable because they remove redundancy and at the same time have problems with peak shift due to more critical timing requirements.

Hard sectoring is another high performance option increasing the storage capacity of a diskette. By adding sector holes tied through the index photosensor to a sector address counter circuit, the need for space

wasting address fields is eliminated, increasing the space available for data. The number of data sectors is increased from 26 to 32 in most hard sectoring schemes.

There have been several complex large scale integration (LSI) chips introduced, such as Nippon Electric's NEC μPD372, which contain most of the circuitry necessary to perform the formatting and device control functions of a floppy disk controller. These chips obviously simplify the task of designing a general purpose, inexpensive controller.

Decoding a diskette's format codes is as far as some controllers go. They leave much work still to be done by software in the host computer in order to retrieve data files from a diskette. At the other extreme some controllers have dedicated intelligence in the form of their own microprocessors. A floppy disk operating system will usually include a general-purpose set of subroutines, called a device handler, which takes care of whatever messy details the controller can't do. A partial list of the tasks required of a device handler for a floppy disk includes:

- determining the track and sector address of a desired file
- doing a seek to the desired track
- scanning sector addresses for a match
- reading, writing and buffering data
- error detection and correction

If the files to be stored are longer than one sector's worth of data, then there must be additional software, usually called a file management package. The file manager determines how to break up and reassemble large blocks of data and maintain a directory of where all the files and remaining free space is located. The file manager is typically used to invoke the device handler in order to keep minor details transparent to the user. The user merely has to call the file manager and ask that a file identified by a name be either written or retrieved. From that point on, he or she can be completely ignorant of the details of lookups, timing, noncontiguous files, error retries and many othe complications required for floppy disk operation. Of course, transparent or not, these two programs add quite a bit of overhead to the task of disk access: the host computer must spend time (up to several hundred milliseconds) between the time a request is made to the file manager and the time the access begins. In most microprocessors and minicomputers, this time is precious and could almost always be spent more profitably elsewhere.

In an effort to reduce unproductive overhead time, many manufacturers are design-

The Dual Sided Floppy

One of the most promising developments in the floppy disk field is the new two-sided floppy. An example of this is the Shugart Associates SA850/851 double sided floppy disk drive shown in photo 2. Details of the drive mechanism are shown in figure 6. The price of the unit is approximately $750, and it is capable of storing up to four times the data of a standard floppy disk drive (1600 K bytes unformatted or 1200 K bytes formatted). The unit is available with double density FM encoding capability (called M^2FM). A metal band driven by a stepper motor is used to position the dual head assembly. Photo 2 shows a closeup of the head assembly.

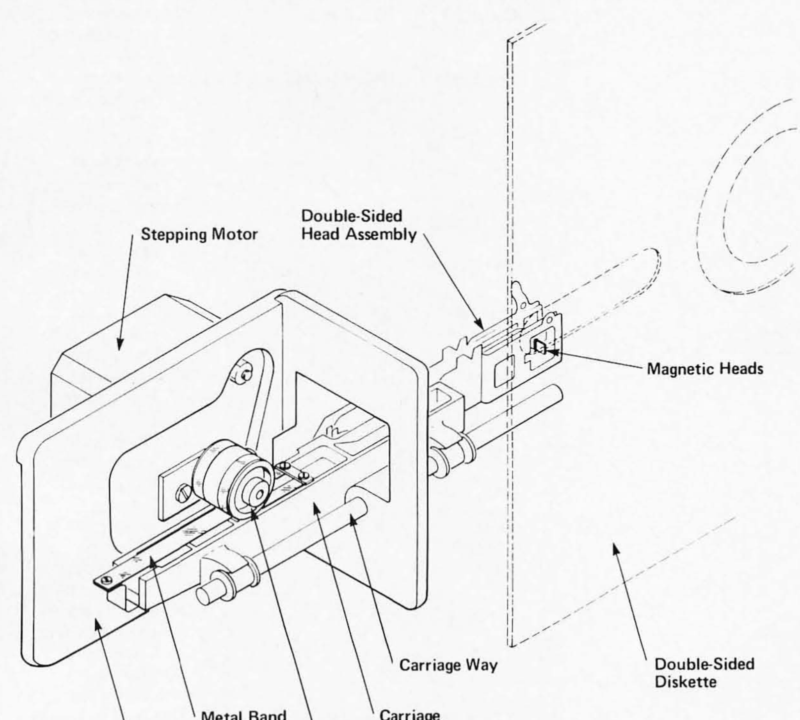


Figure 6: A diagram of the Shugart SA850 dual sided floppy disk drive. Note the metal band which is used for positioning the head assembly. (Graphics courtesy Shugart Associates.)

Command Language

Command	Command Syntax	Function
Allocate	A X n	Create an empty file of length n with name A*
Copy	C X Y	Copy file X into Y
Delete	D X	Delete file X
Eject	E d	Eject diskette in drive d
File	F u X	Open file X and assign it logical unit number u
	F u	Close file associated with u
	F	Close all files
Gap	G d	Compress data gaps on drive d
Input	I t s d	Read sector s on track t on drive d
Kill	K d seq	Initialize diskette on drive d with optional interleaved sector sequence seq
Load	L X	Read entire file X
Name	N X Y	Rename X by Y
Output	O t s d	Write sector s on track t on drive d
Position	P u s byte	Position open file associated with u to the relative sector and byte offset
	P u	Report current position of open file associated with u
Query	Q X	Report index track info for file X
Read	R u n	Read n bytes from open file u
	R u	Read variable length record from open file u
Save	S X	Write a new file called X
Test	T d	Run diagnostics on drive d
Write	W u n	Write n bytes to the open file associated with u

***File References (FR):** A 4 part identifier for each disk file consists of a name, version, type and drive number. Identifiers consist of alphanumeric characters separated by special punctuation characters. "Wild card" and partial file constructions are allowed.

Table 2: The command set for the PerSci intelligent floppy disk controller.

ing and producing "intelligent" controllers, which vary from dumb to very bright. The label "intelligent" does not specify exactly what the controller is capable of, but indicates only that it can reduce the host's workload associated with floppy disk use. These intelligent controllers usually take the form of a separate high speed dedicated microprocessor attached to the host's DMA-IO bus. The host can issue a small number of macrocommands which the controller processor can decode and execute on its own.

There are two paths to intelligence that manufacturers are currently taking. The first, similar to the channel computer concept, is to provide a controller capable of executing user-written mini-instructions directly and independent of the host's main memory. These mini-instructions, above the level of the controller's microcode but below the level of typical user file commands (GET file, PUT file), provide a great deal of flexibility in terms of incorporating the floppy into a preexisting high performance operating system. Controllers of this type might be most efficiently used in high performance minicomputer systems. Here a sophisticated file manager program can take full advantage of the flexibility and other features of this type of controller and at the same time reduce the host's workload. An example of this type of controller is Scientific Micro Systems' FD0300 family of controllers for various different computer families.

The second approach is to incorporate the entire device handler and most of the file manager programs into the controller. This method takes care of the hardware details and most of the software details of file management for the user. However, this is at the expense of eliminating control of file formats, directories and other things, since the controller usually has a fixed and rather simplistic approach to file management. This is not necessarily bad for personal microcomputer systems, in which the goal is not necessarily tricky data file manipulations but rather rapid and convenient access to a small number of files (such as assemblers, compilers, sources and object codes). In fact, this type of controller may be the best bet for meeting that sort of system requirement. An example of this type of controller is the PerSci 1070. The 1070 communicates with its host largely by single character ASCII codes which signify major file operations such as S for seek, D for delete, and G for get (read). The complete list of 1070 op codes is fairly powerful, as can be seen in table 2.

Adding intelligence to a floppy disk con-

troller costs about $200 to $300. This brings the total cost to about $400 to $600, matching the cost of most floppy disk drives. Including another $200 for a power supply, cables and a cabinet raises the cost of a floppy disk system to double that of many personal computer systems. Many personal computer systems (the author's included) are little more than workshop curiosities. The addition of a fast mass storage device would transform them into convenient and powerful interactive tools.■

GLOSSARY

Address mark: A special byte used for formatting data (see figure 5).

Bail: A solenoid operated mechanical device used to lift the pressure pad away from the floppy disk during insertion and removal.

Cartridge: A square plastic sleeve used to protect the floppy disk.

Data field: That portion of a sector in which data can be stored.

Floppy disk: A flexible plastic disk used for bulk data storage and retrieval.

FM encoding: A floppy disk encoding technique in which clock bits alternate with data bits in a serial pulse train.

Gap byte: A data byte consisting of all zeros, used as a buffer between various regions on the floppy disk.

Hard sectoring: A technique of sector identification in which each sector is assigned a unique physical hole on the disk, and 32 sectors can be put in a single floppy disk track. Since this form of sectoring requires a unique hole pattern and a different type of controller, it is incompatible with soft sectored disks.

Head: An electromagnetic device used for reading, recording or erasing data on a magnetic medium such as the floppy disk.

Sector: One of several fixed length subdivisions of a floppy disk track used for storing data and ID information. Typical mechanisms use 26, 32 or a similar small number of sectors per track.

Soft sectoring: A technique of sector identification in which only one physical hole on the disk is used to synchronize the beginning of a track's data. Then, the remainder of the track is divided into sectors which contain the sector identification as part of the fixed format of data on the track. Only 26 sectors per track are possible due to the formatting information which must be recorded with each sector. This format is incompatible with the hard sectored format which achieves higher data content.

Track: One of 77 concentric rings of data on a floppy disk. The radial distance between tracks is 0.0213 inch (0.05 cm).

Technical Forum

Gerry Wheeler
Computer Communications Network Group
University of Waterloo
Waterloo, Ontario CANADA

Undocumented M6800 Instructions

According to Motorola there are 197 valid operation codes for the M6800 microprocessor. This means that of the 256 possible 8 bit combinations, 59 are called invalid instructions.

Have you, like myself, ever wondered about these invalid codes? What would happen if you accidentally executed one? It does happen sometimes, of course, whenever your latest software creation takes an unexpected leap into never never land and begins executing randomly set memory locations. What are those holes in the op code chart anyway?

The mystery of those holes held my attention until the suspense was unbearable. To satisfy my gnawing curiosity I executed those codes deliberately, defying man and Motorola! And I got some interesting results.

Some of those codes seem to be just NOPS: they do nothing. Others change the flags in the condition code register according to some pattern that is, as yet, undeciphered.

But let me tell you about a couple of the interesting ones. See table 1 for descriptions of six instructions that Motorola didn't tell us about. The mnemonics are, of course, assigned by me.

The first one, NBA, is self-explanatory. The A and B accumulators are ANDed together, and the result is stored in A. I had to use NBA as the mnemonic because ABA is already used by Motorola. This instruction has been checked out thoroughly, and seems to be perfect, even setting the condition codes correctly. The only uncertainty is its execution time.

The store immediate instructions may require some explanation. Consider for a moment the load immediate instructions. These instructions take the byte following the op code and put it into the appropriate register. Therefore the store immediate instructions should store the register into the byte immediately after the op code, right? The only flaw is that there is a hole left after the instruction, and the register is stored after that (see figure 1). Note that the next instruction executed is the byte following the newly stored register. This means that the store immediate A and B instructions are three bytes long, and the store immediate X and SP instructions are four bytes long!

Now for the big surprise. This one has been dubbed HCF for Halt and Catch Fire. Well, almost. When this instruction is run the only way to see what it is doing is with an oscilloscope. From the user's point of view the machine halts and defies most attempts to get it restarted. Those persons with indicator lamps on the address bus will see that the processor begins to read all of memory, sequentially, very quickly. In effect, the address bus turns into a 16 bit counter. However, the processor takes no notice of what it is reading. . . it just reads. The only way out of this race is with the RESET line. The machine ignores the IRQ, NMI and HALT lines. For all intents and purposes the processor has halted and caught fire! It is quite possible that the HCF instructions are put into the 6800 design intentionally in the interest of production testing of newly fabricated processor chips.

Table 1: A list of six undocumented M6800 instructions and their definitions. The operations and operation codes which invoke them are defined in the column labelled Result, and the next instruction address is given in each case. Halt and Catch Fire (HCF) does not have a "next instruction" address because the processor hangs up.

Name	Mnemonic	Hexadecimal Op Code	Result	Next Instruction At
AND accumulators	NBA	14	A.B→A	PC + 1
store ACCA, immediate	STAA	87	A→PC+2	PC + 3
store ACCB, immediate	STAB	C7	B→PC+2	PC + 3
store SP, immediate	STS	8F	SPh→PC+2;SPl→PC+3	PC + 4
store IX, immediate	STX	CF	IXh→PC+2;IXl→PC+3	PC + 4
Halt and Catch Fire	HCF	9D or DD	see text	Not applicable

This one instruction might provide the automatic test equipment with a quick initial indication of whether the particular processor chip is a total dud, or a prospect for more detailed automatic testing and verification of defect free operation.

While these instructions are now documented, some warnings must in all fairness be stated lest the user run into problems. The primary warning is that there may be a reason that they were left undocumented: they may not work with every 6800 processor, so any software intended for production, distribution to friends or for publication should *never* use these instructions. At different times during the history of M6800 production at Motorola, revisions and changes in the production masks may alter the effects of these instructions without any warning to users; after all, an undocumented instruction is not there from Motorola's point of view, so why tell the users about changes in its definition? Similarly, when 6800 parts are acquired from suppliers other than Motorola, use of independent designs for the production masks by the second source leaves definition of these undocumented instructions unspecified and not necessarily identical to Motorola's definitions. But these warnings apply only to programs to be distributed in some way; if your personal processor executes these instructions and you find a use for them in your own handcrafted assembly language programs, then by all means take advantage of them.■

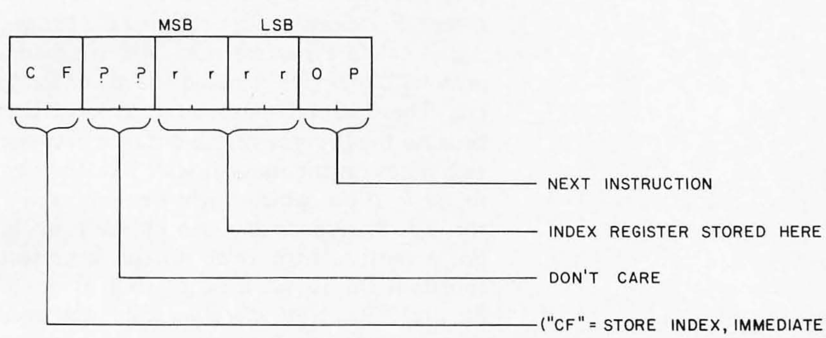


Figure 1: The "Store Index Immediate" instruction requires four bytes of memory, as illustrated here. The operation code hexadecimal CF is followed by one byte which is "don't care" as far as the operation of this instruction is concerned. The third and fourth bytes of the operation receive the 16 bit address value from the index register in the normal order. In this diagram, rrrr is the 16 bit target for the immediate store, and OP is the first byte of the next instruction. Operation of the "Store Stack Pointer Immediate" instruction is similar.

A $19 Music Interface

(And Some Music Theory for Computer Nuts)

Bill Struve
800 Madison Av
Memphis TN 38163

"It's all Relative." So it is in physics as it was in music. About 600 BC Pythagoras discovered that strings under equal tension sounded harmonious if their lengths were in ratios of small whole numbers like 2/1, 3/2, 4/3, 5/3, etc. Many experiments throughout the world since that time have told us that in music, it is the ratios of the frequencies of the notes that count, not the absolute frequencies. It has only been in recent times that there has been international agreement that A above middle C is 440 Hz. Musicians call the "distance" between two notes an *interval*. Musical intervals are actually the ratios of the frequencies of two notes, and are so important in music that many of the ratios, or intervals, have names. For example, 2/1 is called the *octave*, 3/2 is called a *perfect fifth*, 4/3 is called a *perfect fourth*, 5/3 is called the *major sixth*, etc. These names make sense to musicians because they represent the distance between two notes on the musical scale like *do re mi fa sol la ti do*, which might be numbered 1 through 8, respectively. An octave is *do* to *do*, a perfect fifth is *do* to *sol*, a perfect fourth is *do* to *fa*, a major sixth is *do* to *la*, etc. The *pure diatonic* scale was constructed to maximize harmony between notes. This scale has been called the natural scale, and is one of the two most widely used scales in Western music. Many unaccompanied singing groups sing on this scale because it sounds right to them, even though they may not be able to tell you the difference between pure diatonic and tempered diatonic scales. Later you'll see how easy it is for a computer to generate notes on this scale.

Pianos, electronic organs, and synthesizers are all tuned to a slightly different scale, the *equally tempered diatonic* scale. J S Bach (1685-1750) played keyboard instruments and composed music which required changing key signatures (which we'll define by example later in this discussion), during the performance. But changing key signatures on an instrument tuned to the pure diatonic scale usually required retuning the instrument as you'll see in a moment. Bach found his way out of this dilemma by slightly mistuning his instruments, a technique which had recently been developed in Europe. This tempering was done so that all key signatures were equally out of tune, or equally tempered. When this is done, the ratio of frequencies of any two adjacent notes turns out to be the twelfth root of two (the value 1.0594631 noted mathematically as $\sqrt[12]{2}$ or calculated in FORTRAN-like languages as 2**(1.0/12)). He chose this ratio because there are twelve half steps per octave and the octave is a ratio of 2/1. Only the octave is kept purely harmonic in this scale: The perfect fifth is 0.11 percent low, the perfect fourth is 0.11 percent high, the major sixth is 0.91 percent high, etc. Since the most discriminating ear can only perceive differences in frequency when they are more than 0.2 percent, the most harmonious intervals (the octave, the fifth and fourth) are indistinguishable between the two scales. But what Bach and the world gained by giving up a little harmonic perfection was a quantum jump in the versatility of fixed tuned instruments (and an added quantum jump in the time and skill required to properly tune one).

FIRST EASY WAY TO BUY A MINICOMPUTER



MINI 12 SPECIFICATIONS

Word Length	12 Bits
CPU	IM 6100 fully static CMOS device
Instruction Set	Identical to the Digital Equipment Corporation PDP-8E
Clock Rate	4MHZ
Major State Time	500NS
Serial interface	20 MA current loop standard, RS-232 Optional
Baud Rate	110 Standard, Others optional
Memory	8192 words standard, expandable to 32 K words
Control Panel	PDP-8E compatible, with additional functions
Parallel Interface	12 input and 12 output lines
Real Time Clock	Programmable, from 10 MS to 40.95 seconds
Counter	Counts External Events
Expansion Bus	50 line, TTL compatible terminated bus structure
Binary Loader	ROM resident.
Monitor Bootstrap	ROM resident.
Power Requirements	100/120/200/240 VAC, 50/60 HZ
Dimensions	2" high x 13" wide x 14" deep



BUT

HOW'M I GONNA PAY FOR IT?

TLF offers 3 purchase plans:

1) CASH with order - and receive a BONUS CERTIFICATE worth $100 on selected MINI 12 accessories.
2) Send $350 with order and pay balance of $545 when ready to ship or COD to postman.
3) NO INTEREST EASY PAYMENT PLAN

HERE'S HOW IT WORKS:

STEP 1

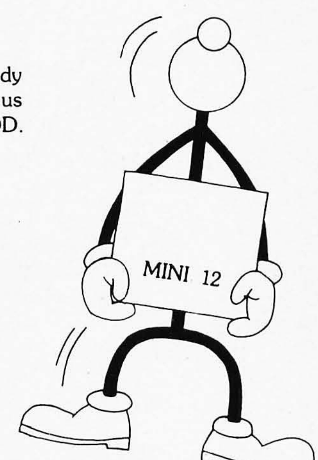
Fill in coupon below and mail TODAY with your check for **$195** to:

TLF
P.O. Box 2298
Littleton
Colorado 80161



STEP 2

When your MINI 12 is ready to be shipped you send us $200 or pay postman COD.



STEP 3

The balance of $500 is paid in 4 equal monthly installments of $125 each.

TOTAL PRICE ONLY $895

WHAT COULD BE SIMPLER?



You can use your BAC/VISA, Master Charge or American Express too!

Make all cheques payable to TLF Corporation.

And your wife will be happy too! In fact she might just buy it herself (for you of course) for Christmas.

Please enter my order for ________ MINI 12 Computers @ $895 each as per the plan checked below:

1 ☐ Enclosed is $895 - I want the Bonus certificate. ☐ Send more information please.
2 ☐ Enclosed is $350 - I will pay $545 when you're ready to ship or COD when delivered.
3 ☐ Enclosed is $195 - I will pay $200 when you're ready to ship and 4 monthly payments of $125 ea.

___Check ___Money Order ___BAC/VISA ___Master Charge ___American Express
Card # ________ Interbank # ________ Exp Date ________
Signature ________
Name ________ Phone ________
Address ________
City ________ State ________ Zip ________

TLF Corporation P.O. Box 2298 Littleton Colorado 80161
Telephone 303 922 6241 Telex 454541
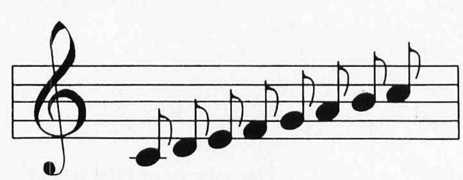
Circle 121 on inquiry card.

Harmonious Computers

Microcomputers can give us both perfection and versatility. Since division by small whole numbers is trivial with digital electronics, it is at first sight more practical to use the pure diatonic scale when digitally generating music, just as it has been more practical to use the equally tempered diatonic scale for music performed on classical keyboard instruments. Changing key signatures in computer generated music is no problem, since the entire instrument may be "retuned" in a few microseconds.

The greatest advantage of the microcomputer is the ease with which anyone can produce music. Years of time consuming practice are not required. Application of computers to music may change music from an activity primarily dominated by motor skills to one dominated by the intellect. Composers no longer have to be skilled at playing an instrument in order to work out their compositions.

Do, re, mi, fa, sol, la, ti, do! North American, English, and Italian children all learn how to sing the scale. Most of them also learn other representations of the same musical scale like: C D E F G A B C, and:



Rarely if ever are any of these youngsters exposed to: 264 Hz, 297 Hz, 330 Hz, 352 Hz, 396 Hz, 440 Hz, 495 Hz, 528 Hz, or to: 1/1, 9/8, 5/4, 4/3, 3/2, 5/3, 15/8, 2/1. These two sets of numbers are also representations of *do, re, mi, fa, sol, la, ti, do* in the pure diatonic scale. Equally valid (especially for the piano) representations of this simple *do* to *do* musical scale are; 261.6 Hz, 293.7 Hz, 329.6 Hz, 349.2 Hz, 392.0 Hz, 440 Hz, 493.9 Hz, 523.3 Hz, which are related to each other by powers of the twelfth root of 2: $2^{0/12}$(=1.000), $2^{2/12}$(=1.1225), $2^{4/12}$, $2^{5/12}$, $2^{7/12}$, $2^{9/12}$, $2^{11/12}$, $2^{12/12}$ (=2.000). As you may have guessed by now, these last two sets of numbers are the frequencies and frequency ratios of the equally tempered scale of *do* to *do* played on a piano.

So far, so good, but if you are as fast as I am at absorbing this material, by now it should be as clear as mud! Organization of facts into a pattern often does wonders for the intellect, so let us organize all this information into one table (table 1) and call it the "Key of C Major" so that musicians will think we are talking about music instead of computers.

You should notice a couple of things about table 1. First, at the bottom line you'll see that I've added a new concept: the musician's idea of *step size*. The steps come in two sizes, whole and half. Remembering that everything is relative, we can talk about step size in terms of the ratio of the frequencies of the pitches, or notes. In the pure scale, a half step up in pitch is an increase of 16/15 in frequency and a whole step up is an increase of 9/8 or 10/9. In the tempered scale all half steps up are an increase in frequency by the twelfth root of two ($2^{1/2}$), and all whole steps up in pitch increase the frequency by the sixth root of two ($2^{1/6}$) which is two half steps:

$$2^{1/12} \times 2^{1/12} = 2^{2/12} = 2^{1/6}$$

Secondly, you should note that the difference between the pure and tempered notes is imperceptible for four of the eight notes. You may be wondering why 440/440 = +.91 percent instead of 0 percent and why 261.6/264 = 0 percent instead of −.91 percent. To answer this, look at the "Frequency Ratio to C" lines and recall that everything is relative so: C(tempered)/C(pure) = 1/1, or 0 percent and A(tempered)/A(pure) = $2^{3/4}/(5/3)$ = 1.6818/1.6667, or +.91 percent.

To make this last point clear let's make a *do* to *do* scale from A = 220 Hz to A = 440 Hz, table 2. I could have made C(tempered) = C(pure), but that would violate an international agreement about A = 440 Hz! Besides, this way I can tell you about a scale in the minor mode. We'll impress the musicians looking over our shoulders by calling table 2 "Key of A Minor."

The two major differences between these two keys are the beginning note and the sequence of whole (W) and half (H) steps up the scale. Both the starting place and the sequence are specified in the name of the key. The key of C major begins with C and proceeds in the *major mode* sequence of steps, WWHWWWH. The key of A minor

Article continued on page 54

Text continued on page 58

A PROFESSIONAL COMPUTER TERMINAL FOR THE SERIOUS HOBBYIST

THE AJ 841 I/O—A COMPLETELY REFURBISHED IBM SELECTRIC TERMINAL WITH BUILT IN ASCII INTERFACE—JUST $995

Features:

- ASCII code
- 14.9 characters per second printout
- Special introductory price—$995 (regularly $1195). 75% discount over original price of new unit.
- Choice of RS 232 Serial Interface or Parallel Interface (requires 3P + S)
- Order direct from factory
- 30 day warranty—parts and labor
- Nationwide service locations
- High quality selectric printing
- Reliable, heavy duty selectric mechanism
- Off line use as typewriter



AJ 841 WARRANTY AND SERVICE IS AVAILABLE IN THE FOLLOWING CITIES:

Los Angeles	Cincinnati
Philadelphia	Detroit
Hackensack	Dallas
Columbus	Houston
Cleveland	Atlanta
San Jose	Chicago
Boston	New York
Washington, D.C.	

ANDERSON JACOBSON

Anderson Jacobson, Inc., 521 Charcot Avenue
San Jose, California 95131, (408) 263-8520

HOW TO ORDER AN AJ 841 I/O TERMINAL

1. Make cashier's check or money order payable to: **ANDERSON JACOBSON, INC.**

   Address your request to:

   Personal Computer Terminal
   ANDERSON JACOBSON, INC.
   521 Charcot Avenue
   San Jose, CA 95131
2. Upon written notification, pick up your terminal at the AJ service office located in one of the above cities. Allow six to eight weeks for delivery.
3. A final check of your unit will be made at the local AJ service office at time of pick up.
4. For warranty or repair service, return unit to designated service location.

CLIP AND MAIL WITH ORDER

SELECT EITHER:
☐ RS 232 Serial Interface
☐ Parallel Interface (requires 3P + S)

Number of units ____________ @ $995. each $____________

Local Sales Tax $____________
Shipping and handling $35.00 each
(excluding San Jose) $____________

TOTAL $____________

NAME ____________

ADDRESS ____________

CITY ____________ STATE ________ ZIP ________

PHONE (______) ____________

FROM THE INVENTORS OF SEX SOMETHING NEW, THE WORLD'S BEST KEPT SECRET.

> ## SEX
>
> Now what self-respecting company resorts to a cheap trick like this to get your attention.
>
> ### WE DO
>
> And it works, doesn't it? So does our new 16K STATIC RAM memory board. It is fully assembled and tested and plugs into your SWTPC 6800 microcomputer—allowing you to expand your system to 48K. The PRICE?? That's the sexy part. It's so BARE we can only reveal it through the mail. (Sent in plain brown wrapper so as not to offend a non-computer oriented spouse.)
>
> Name ____________
>
> Street ____________
>
> City ____________
>
> State & Zip ____________
>
> **SMOKE SIGNAL BROADCASTING**
> P.O. Box 2017
> Hollywood, California 90028

Exactly one year ago, we introduced SEX to BYTE Magazine with this little ad. Your response was tremendous — much greater, in fact, than to any of our other larger ads which made no mention of sex. Researchers at the psychology department of TJU say that, because the readers of BYTE are not a controlled group (some irresponsible parents even let their children read the magazine), no statistically valid conclusions can be drawn. We believe, however, that we may have stumbled across something of even more interest to computerists than the old 8080. It is, in fact, the world's best kept secret.

What is it you ask? First, please keep this confidential. If word got out to too many people, it might depress the market for late model 8080's so much that one of those high priced units could be purchased for less than a comparably equipped new economical 6800 with automatic bootstrap and power disc drives. The world's greatest secret is: NOW THERE IS A LOT OF GOOD SOFTWARE FOR THE 6800. For instance, there are two disc based editors and three assemblers available for the SMOKE SIGNAL BROADCASTING BFD-68 disc system. A disc based BASIC COMPILER with data file capability is available and so is a BASIC INTERPRETER with data file capability. We also have a DISASSEMBLER with TRACE capability, a SOURCE GENERATOR and a smart 1K MONITOR program with I/O, tape load and punch, breakpoint, single step capability and much more. If you're a newcomer to computing and want to learn what BASIC is all about, we have a LEARN BASIC package that leads you from basic BASIC to advanced BASIC in 12 easy lessons. All this plus the best microcomputer disc operating system around.

SUPER EDITOR: The SE-1 is a content oriented editor with string search and block move capability. Changes may be made by referring to line number or string content or a combination of references. Naturally, it is designed for file transfers to and from the BFD-68 so the size of the edited file is limited only by the capacity of the diskette. $29 on diskette or cassette.

SUPER ASSEMBLER: The SA-1 inputs source code from a file on the BFD-68 and outputs object code to disc file. Some very large programs can be assembled with the SA-1 since the source code resides on disc and is not resident in memory. Assembly listings include alphabetized and tabulated symbol table: $29 on diskette or cassette.

Complete source listings are included for both the SE-1 and SA-1. Order both for $53 and save $5.

SMARTBUG — A CURE FOR MIKBUG!TIS: A super smart Motorola-Mikbug replacement that preserves almost all Mikbug entry locations so your present programs will run without modification. Uses ACIA for baud rates to 19,200 and includes many additional features including a software single-step trace command. Manual and source listing for $19.50.

TRACE-DISASSEMBLER: The TD-1 in the trace mode allows you to trace through a program and monitor and change the registers or memory as you go along. In the disassembler mode, it provides you with a way to make a program listing from a program when you only have the object code. $19.95 on cassette. Add $5.95 for diskette.

SOURCE GENERATOR: The SG-1 is a very fancy disassembler that takes object code in memory and creates a listing and outputs source code to cassette tape or the BFD-68 in either SWTPC CO-res or SSB Editor format. The source code generated includes labels so that the source file created may be edited and re-assembled with whatever changes the user wishes to make. $24.95 on cassette. Add $5.95 for diskette.

BASIC COMPILER: The Software Dynamics Basic Compiler has been adapted for use on the BFD-68. This is a super fast business oriented BASIC with PRINT USING statement, disk data file capability and ten digit accuracy. Because it is a compiler, you can develop business programs, compile them and deliver an object module, thus, keeping the BASIC source program as your trade secret. Write for more information.

LEARN BASIC is a 12 lesson programmed course in BASIC. It takes the new owner of a 12K or larger 6800 computer system from beginning BASIC to advanced BASIC in easily understood steps. Requires SWTPC 6800 version 2.0 BASIC. Includes 65 page manual. $39.95 on cassette or diskette.

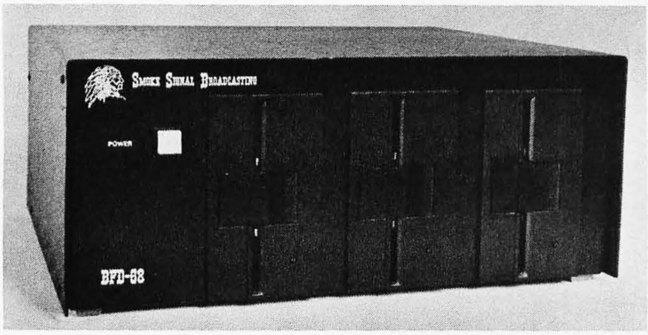
A BASIC INTERPRETER for the BFD-68 with data file capability is available for $39.95 from Computerware, 830 First Street, Encinitas, CA 92024. Write them for more details.

SMOKE SIGNAL BROADCASTING

P.O. Box 2017, Hollywood, CA 90028 • (213) 462-5652

Circle 104 on inquiry card.

FULLY ASSEMBLED PRODUCTS FOR THE TRULY COMPATIBLE SS-50 6800 BUS.



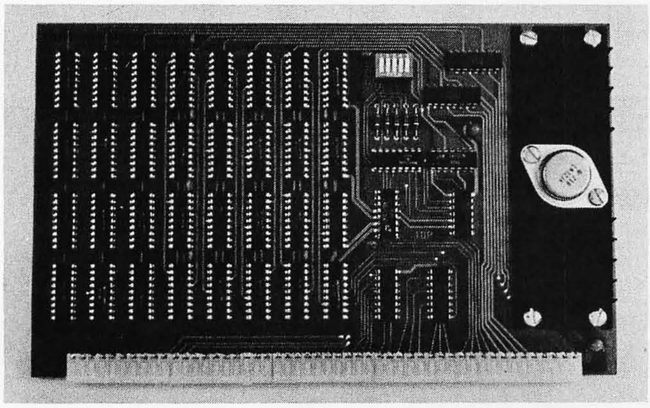
Our Basic Floppy Disc System (BFD-68) must, in all modesty, be called superb. It comes completely assembled with a disc controller that is plug compatible with the SWTPC 6800. The cabinet and power supply are capable of handling up to 3 Shugart Mini-Floppy Drives. One drive is included in the price of the BFD-68 and others may be added easily at any time. Or you may save money by ordering the dual-drive BFD-68-2 or triple drive BFD-68-3 (pictured). Price: BFD-68 **$795,** BFD-68-2 **$1139,** BFD-68-3 **$1479,** SA-400 Drive **$360.**

The BFD-68 includes our Disc Operating System Software. The software provides direct commands to name and rename files, transfer memory to disc and disc to memory and to automatically jump to the starting location of any program loaded from disc to memory. The direct command names are: RUN, GET, GETHEX, CLOSE, SAVE, DELETE, APPEND, RENAME, COPY, LIST, LINK and PRINT. In addition, the Disc File Management subroutines are available to create files under your program control.

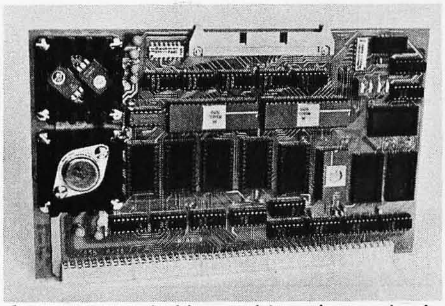
A bootstrap PROM is included on the controller board to initiate the Disc Operating System which loads into a 4K memory board located at 7000 or optionally at D000. Thus, you can be up and running from a cold start in just a few seconds.

SUPER SOFTWARE: Free patches are provided for SWTPC BASIC version 2.0 and Co-Resident Editor/Assembler. These patches allow the SAVE and LOAD commands to work with the disc or the cassette at your option.

See the opposite page for more details on the Super Software available for the BFD-68.



M-16-A: The M-16-A is a single power supply fully STATIC 16K memory system. It is fully buffered and requires only half the power of a similar size system using low power 2102's. With the M-16-A, you can expand your system to 48K, add one of our EPROM boards, our BFD-68 disk controller board and still have room to spare. The M-16-A is switch selectable to any 4K starting address and hardware write protect is included. Price **$529.**



P-38: The P-38 series modules are available in 3 configurations. The basic P-38 is an 8K EPROM board containing room for 8 2708's. Like all our products, it is completely assembled. As a bonus, one 2708 is included which contains SMARTBUG, our 1K Mikbug compatible monitor program. The P-38 is switch selectable to any 8K location. Price **$179.**

The P-38-I contains all the features of the basic P-38 plus a built in interface to the POP-1 and the Oliver Paper Tape Reader. Price **$229.**

The P-38-FF is a plug-in interface card for the ICOM Frugal Floppy and the other ICOM full size floppy disks. It contains all the features of the P-38-I plus an additional 2708 with the ICOM bootstrap software. ICOM's 6800 FDOS-II is included on diskette. Price **$299.**

The **PS-1** power supply kit provides plus and minus 16 volts required for the P-38 series boards. Also, it allows a wiring modification to be made to the 8 volt supply that will increase its output by one volt. Price **$24.95.**

NEW PRODUCT: The POP-1 is a 2708 EPROM programmer that is contained in a separate cabinet outside the 6800 and connected by ribbon cable. The POP-1 uses a separate self-contained power supply for the programming voltage. There is plenty of power to program 2708's from any manufacturer — no need to choose only the most expensive 2708's. The POP-1 interfaces to the SMOKE SIGNAL BROADCASTING P-38-I EPROM board and complete software is provided on cassette. The software allows you to duplicate an existing 2708 (making changes if you wish) or to transfer a block of RAM to EPROM. An adaptive programming technique is used that allows most 2708's to be programmed within 15 seconds. Instructions are also provided showing how to modify the POP-1 to program the TMS 2716. Price **$129.**

ALL OUR PRODUCTS EXCEPT THE PS-1 ARE COMPLETELY ASSEMBLED.

MASTERCHARGE AND VISA CARDS WELCOME.

SMOKE SIGNAL BROADCASTING

P.O. Box 2017, Hollywood, CA 90028 • (213) 462-5652

Circle 104 on inquiry card.

Continued from page 50

Key of C Major

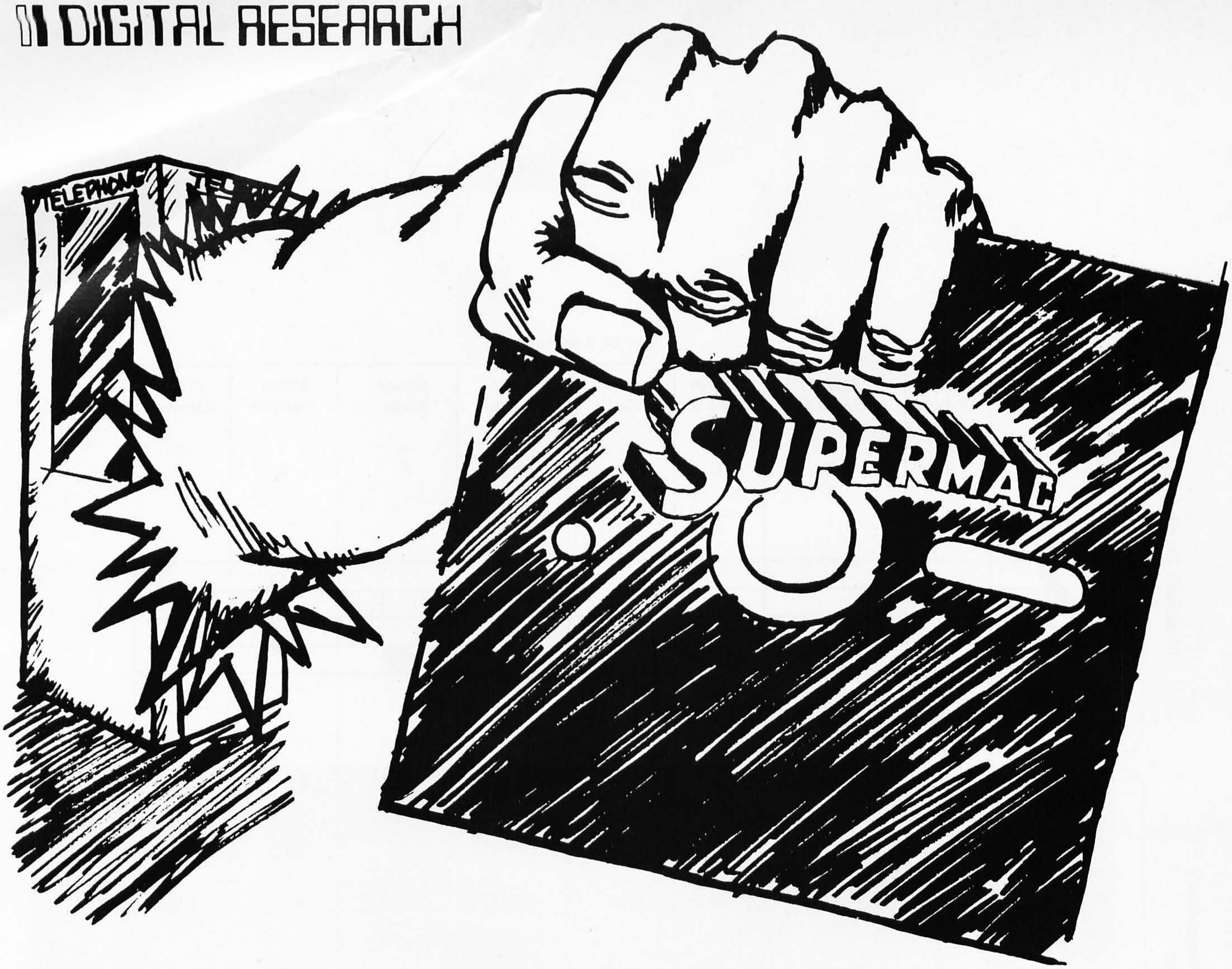
Musician's Terms									
Interval From C:	Unison	Major Second	Major Third	Perfect Fourth	Perfect Fifth	Major Sixth	Major Seventh	Octave	
Sung:	*do*	*re*	*mi*	*fa*	*sol*	*la*	*ti*	*do*	
Called:	C	D	E	F	G	A	B	C	
Written:									
Physicist's or Mathematician's Terms									**Units**
Frequency									
Pure:	264	297	330	352	396	440	495	528	Hz
Tempered:	261.6	293.7	329.6	349.2	392.0	440	493.9	523.3	Hz
Frequency Ratio to C for,									
Pure:	1/1	9/8	5/4	4/3	3/2	5/3	15/8	2/1	Hz/Hz
Tempered:	1	$2^{1/6}$	$2^{1/3}$	$2^{5/12}$	$2^{7/12}$	$2^{3/4}$	$2^{11/12}$	2	Hz/Hz
Tempered/Pure:	0	−0.23	+0.79	+0.11	−0.11	+0.91	+0.68	0	%
Musician's Terms									
Step Size:		Whole	Whole	Half	Whole	Whole	Whole	Half	

Written:
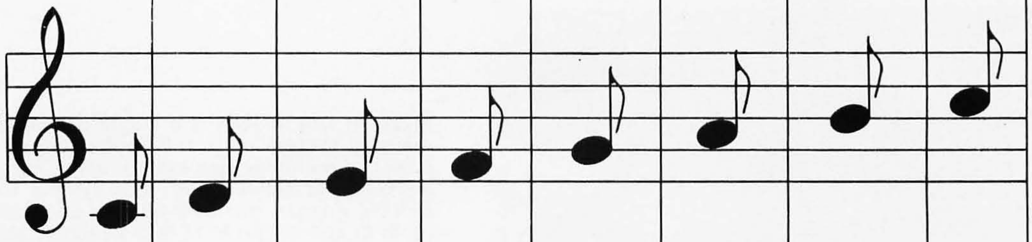


Table 1: The key of C major. There is a direct equivalence between a musician's terminology for musical concepts and the physicist's or mathematician's precise measures of the idea. One of the attractions of music is this low level precision involved in the creation of high level emotional sensations.

DIGITAL RESEARCH



NEW CP/M™ SOFTWARE

Digital Research is pleased to announce the availability of our new macro assembler called **"MAC"** which is upward compatible from our previous 8080 assembler provided with version 1.3 of CP/M™. Our new assembler is compatible with the latest Intel macro standard, and incorporates several facilities which will prove quite useful:

IF, ENDIF, ELSE provide conditional assembly facilities which are controlled by boolean expressions involving the arithmetic operators (+, -, *, /, MOD, and unary -), shift and mask operations (SHL, SHR, AND, OR, XOR), and relational operators (LT, LE, EQ, GE, and NUL).

MACRO definitions allow groups of instructions to be stored and substituted in the source program, as the macro names are encountered. Definitions and calls can be nested, symbols can be constructed through concatenation (using the special & operator), local symbols can be created (using the LOCAL pseudo operation). Macro parameters can be formed to pass arbitrary strings of text to a specific macro for substitution during expansion. In particular, the **MACLIB** (macro library) feature allows the programmer to define a particular set of macros for generation of machine code for any specific 8 or 16 bit machine which does not match the Intel 8080 instruction set. Macro libraries are included with **MAC** for the Zilog Z-80 instruction set, along with macros for performing simple peripheral and sequential file I/O under CP/M.

IRPC, IRP, REPT provide repetition of source statements under control of a list of characters or items to be substituted each time the statements are expanded. This feature is particularly useful in generating groups of assembly language statements with similar structure.

SORTED SYMBOLS are provided in a diskette file suitable for listing on your line printer or use during debugging.

TITLE, PAGE are provided to control page ejects and titles on each page of the source listing.

PARAMETERS can be specified when MAC is started to control the source and destination of particular files as well as listing formats for macro generation.

MAC DOCUMENTATION

MAC documentation includes the **"Macro Assembler Language Method and Applications Guide"**, the most complete guide to macro applications available today. Examples include macro-based languages, high-level control structures and operating system interfaces. Over 60 listings of working 8080 macro assembly language programs are provided so this manual can be used as a study guide in advanced macro applications. (The manual is available for separate purchase.)

The macro assembler occupies a 12K region of memory, and requires concurrent operation with CP/M system, thus requiring 16K of program space during operation. Additional memory space is used for the symbol table, resulting in a minimum usable memory size of about 20K.

***PLEASE NOTE:** MAC is a copyrighted product of Digital Research and may not be copied for use by any individual other than the purchaser. Due to the proprietary nature of MAC, the machine code diskette is available **only** to registered owners of the CP/M™ Operating System. Although you may purchase the "Macro Assembler Language Manual and Applications Guide" separately, Digital Research **cannot** accept diskette orders without your registered CP/M™ Serial Number. (If you haven't returned your CP/M™ Registration Card, you may do so at this time.)

ORDER FORM:

I would like to purchase:

( ) **"MAC Macro Assembler Language Manual and Applications Guide"** Only (200 pages) for **$15.00**

( ) **"MAC Macro Assembler Language Manual and Applications Guide"** and **MAC Machine Code Diskette*.** CP/M™ Serial No. ________ for **$70.00.**

( ) **MAC Machine Code Diskette*** Only. CP/M™ Serial No. ________ for **$60.00.**

California residents please add 6% sales tax.

Total purchase: $ ________

( ) Please charge to my
Mastercharge # ________
BankAmericard # ________
Expiration Date ________
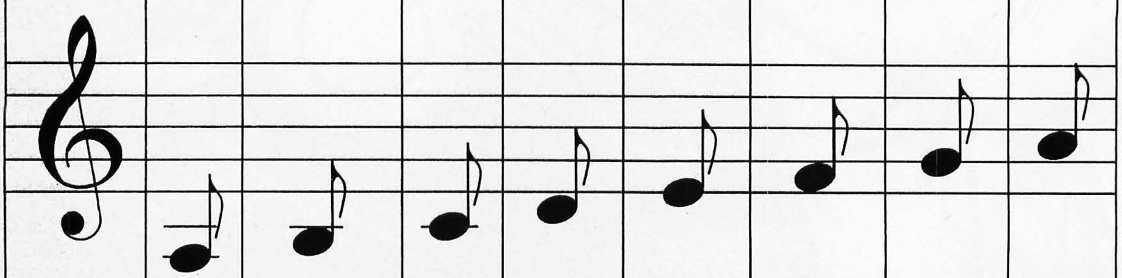
( ) I would like further information on:
( ) CP/M™ Operating System.
( ) MAC Macro Assembler.

Name ________
Address ________
City ________ State ______ Zip ______

DIGITAL RESEARCH
P.O. Box 579, Pacific Grove, CA 93940

Circle 39 on inquiry card.

Key of A Minor

Interval From A	**Unison**	**Major Second**	**Minor Third**	**Perfect Fourth**	**Perfect Fifth**	**Minor Sixth**	**Minor Seventh**	**Octave**	
Sung*	*do*	*re*	*mi*	*fa*	*sol*	*la*	*ti*	*do*	
Called	A	B	C	D	E	F	G	A	
									**Units**
Pure Frequency	220	247.5	264	293.3	330	352	391.1	440	Hz
Tempered Frequency	220	246.9	261.6	293.7	329.6	349.2	392.0	440	Hz
Pure Ratio To A	1/1	9/8	6/5	4/3	3/2	8/5	16/9	2/1	Hz/Hz
Tempered Ratio To A	1	$2^{1/6}$	$2^{1/4}$	$2^{5/12}$	$2^{7/12}$	$2^{2/3}$	$2^{5/6}$	2	Hz/Hz
Tempered/ Pure	0	−0.23	−0.90	+0.11	−0.11	−0.79	+0.23	0	%
Pure A Minor/ Pure C Major	0	0	0	−1.25	0	0	−1.24	0	%
Step Size		Whole	Half	Whole	Whole	Half	Whole	Whole	

Musician's Terminology (Interval through staff); Physicist's or Mathematician's Terminology (Pure Frequency through Pure A Minor/Pure C Major); Musician's Terminology (Step Size)

*North American and English children learn a movable *do* scale, so *do* can be any note. The French and Italians have a fixed *do* system so *do* is C.

Table 2: The key of A minor. As in table 1, we note the same information, but start the scale on A instead of C. This changes the order of half and whole steps (bottom line) from a major mode sequence to a minor mode sequence; an extra line has been added to show the frequency ratios of the minor key with respect to the major key.

Computer Mainframe System

First in the TEI family . . . The MCS-112 and 122 Mainframe Systems.

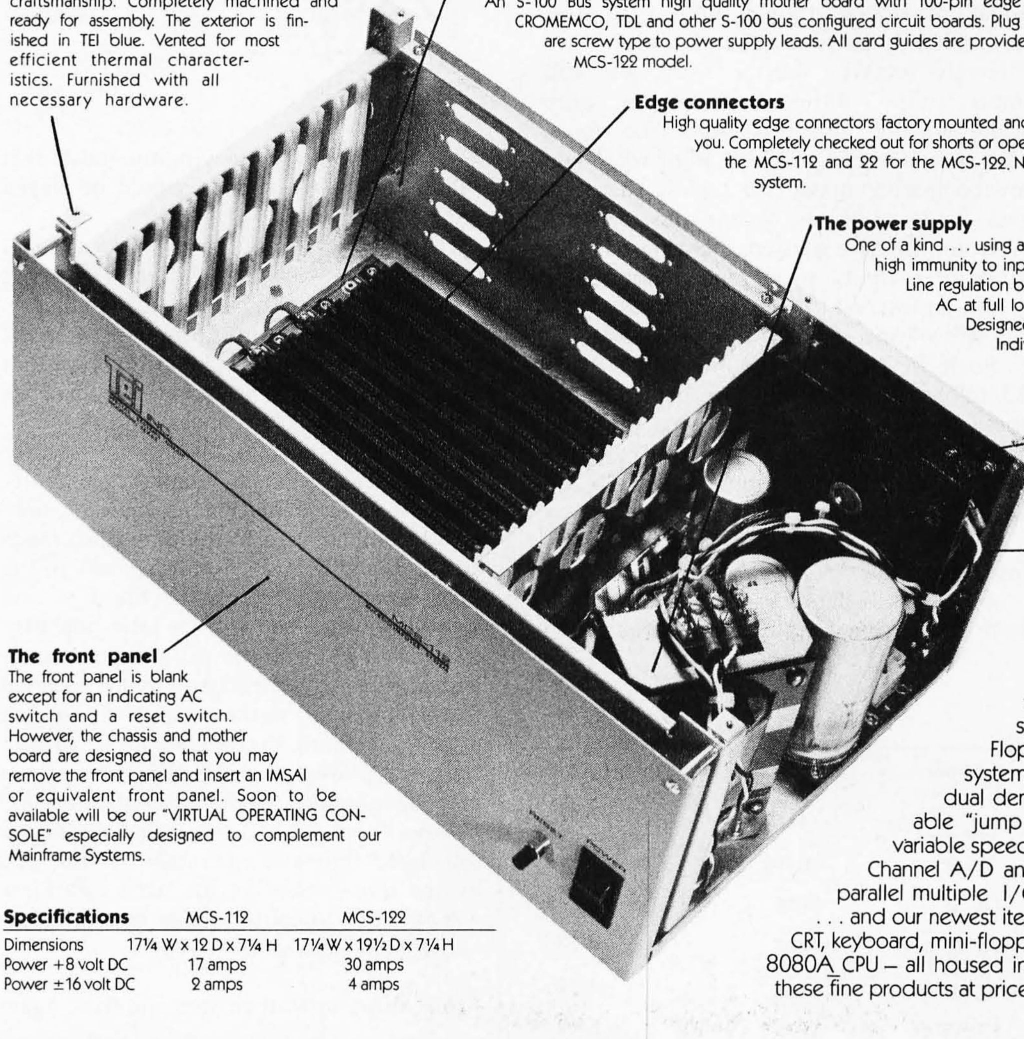
"The Base on Which to Build"

The cabinet
A heavy duty, precision formed cabinet of fine craftsmanship. Completely machined and ready for assembly. The exterior is finished in TEI blue. Vented for most efficient thermal characteristics. Furnished with all necessary hardware.

The motherboard
An S-100 Bus system high quality mother board with 100-pin edge connectors. Compatible with IMSAI, MITS, CROMEMCO, TDL and other S-100 bus configured circuit boards. Plug connections for reset switch. Voltage terminals are screw type to power supply leads. All card guides are provided. 12 slots for MCS-112 model and 22 slots for MCS-122 model.

Edge connectors
High quality edge connectors factory mounted and wave soldered to eliminate this nuisance for you. Completely checked out for shorts or open traces. ALL edge connectors furnished, 12 for the MCS-112 and 22 for the MCS-122. No additional expense when you expand your system.

The power supply
One of a kind . . . using a constant voltage transformer (CVT) with a very high immunity to input line noise . . . greater than 100 db rejection. Line regulation better than ± 1% from an input of 95 to 140 Volt AC at full load to 85 to 140 Volt AC at three quarter load. Designed to meet UL-478 specifications (EDP SPECS). Individual fusing on all input and output voltage lines. See specifications below for power ratings.



The cooling system
A 115 CFM muffin fan with a commercial grade washable filter will provide clean airflow over all circuitry.

The wiring
All wiring is color coded and ALL is precut to length with connecting lugs factory machine applied. Soldering is held to an absolute minimum.

The front panel
The front panel is blank except for an indicating AC switch and a reset switch. However, the chassis and mother board are designed so that you may remove the front panel and insert an IMSAI or equivalent front panel. Soon to be available will be our "VIRTUAL OPERATING CONSOLE" especially designed to complement our Mainframe Systems.

Specifications	MCS-112	MCS-122
Dimensions	17¼ W x 12 D x 7¼ H	17¼ W x 19½ D x 7¼ H
Power +8 volt DC	17 amps	30 amps
Power ±16 volt DC	2 amps	4 amps

NOW . . . TEI puts it all together for you. Mainframe systems (6, 12 and 22-slot) . . . Floppy and mini-floppy disc drive systems (single, double and triple with dual density) . . . a Z-80 CPU with addressable "jump to" and autostart capability (also variable speed control) . . . 8K and 16K RAM . . . 16 Channel A/D and D/A converter . . . 3 serial + 3 parallel multiple I/O . . . and other supporting boards . . . and our newest item, the PROCESSOR TERMINAL — A CRT, keyboard, mini-floppy disc and 12-slot mainframe with a 8080A CPU – all housed in one quality aluminum case. All of these fine products at prices you will like. **Watch for them all.**

Contact your local TEI dealer or if you are not near one of our dealers, write or call CMC MARKETING CORP direct for more information.
(DEALER INQUIRIES INVITED)

PLEASE SEND ME:
☐ MCS-112 Kit @ 395.00 ☐ MCS-112 Assembled @ 445.00
☐ MCS-122 Kit @ 495.00 ☐ MCS-122 Assembled @ 575.00
Texas residents add 5% Sales Tax
I Enclose Check ☐ or Money Order ☐
CMC MARKETING CORP
7231 Fondren Rd, Houston, TX 77036 Telephone (713) 774-9526

TEI INC.

Circle 20 on inquiry card.

Text continued from page 50

starts with A and proceeds in the *minor mode* sequence, WHWWHWW.

Look at the frequencies of the notes called D and G in these two keys. For tempered tuning, each of these notes keeps the same frequency although the key changes from C major to A minor. For pure tuning, however, each of these notes must be lowered by 1.25 percent when changing from C major to A minor. A singer or violinist does this during a performance, but can you imagine a pianist or organist stopping in the middle of a performance to retune two notes in each octave? Bach's equally tempered tuning survives all such key shifts quite well. The most sensitive intervals (octave, fourth, fifth) are still imperceptibly different from the pure scale, and the other intervals get no worse. You should notice one more thing when you are comparing these two tables. There are two kinds of thirds, sixths, and sevenths. As you may have guessed, there are also two kinds of seconds, major and minor. There is also an interval called the *tritone*, so there can be twelve equal half steps per octave.

So if we list all of the intervals, we find 13 to get 12 half steps per octave. Since these thirteen intervals form what is known as the *chromatic scale*, we'll call this list "Intervals of the Chromatic Scale" and write it down in table 3.

You can learn at least five things by inspecting table 3.

First, the ♭ sign is used to denote a half step down from a note and is called a *flat*. The sign for a half step up is # and is called a *sharp*.

Second, you should now be able to write the notes used in the scales of all major and minor keys. For example, the key of C minor begins with C and proceeds WHWWHWW, so it would be: C, D, E♭, F, G, A♭, B♭, C. The *key signature* is the shorthand used by musicians to specify the key at the beginning of each line of music:


= KEY OF C MINOR

This tells the person playing the music that all of the Es, As, and Bs should be played one half step flat.

Third, the major and minor modes sound different because different intervals are used for the third, sixth, and seventh.

Fourth, the two most dissonant intervals, the minor second and the tritone, are not used in any major or minor key, but are needed for some key changes.

Fifth, and perhaps most important for implementation on a "dinky" computer and for experimentation, is that the only prime numbers used in the pure pitch ratios are 2, 3, and 5. Also, 5 only appears to the first power and 3 only to the first and second powers. You will see later how easy it is to implement the pure diatonic scale with inexpensive integrated circuits external to the computer, so the computer is not tied up by generating the pitches itself. In contrast, the powers of the twelfth root of two may be obtained from the moderately expensive "top octave" integrated circuit, or calculated (but not accurately) in real time by the dinky itself. In the latter case there will be little computer power left for calculating the melody or harmony.

	Interval	C Major	A Minor	Pure Ratio	Tempered Ratio
1	Unison	C–C	A–A	1/1	$2^{0/12}$
2	Minor Second	C–D♭	A–B♭	16/15	$2^{1/12}$
3	Major Second	C–D	A–B	9/8	$2^{2/12}$
4	Minor Third	C–E♭	A–C	6/5	$2^{3/12}$
5	Major Third	C–E	A–D♭	5/4	$2^{4/12}$
6	Perfect Fourth	C–F	A–D	4/3	$2^{5/12}$
7	Tritone	C–G♭	A–E♭	(64/45 or 45/32)	$2^{6/12}$
8	Perfect Fifth	C–G	A–E	3/2	$2^{7/12}$
9	Minor Sixth	C–A♭	A–F	8/5	$2^{8/12}$
10	Major Sixth	C–A	A–G♭	5/3	$2^{9/12}$
11	Minor Seventh	C–B♭	A–G	16/9	$2^{10/12}$
12	Major Seventh	C–B	A–A♭	15/8	$2^{11/12}$
13	Octave	C–C	A–A'	2/1	$2^{12/12}$

Table 3: Intervals of the chromatic scale.

From Music to Mathematics and Back Again

Webster defines *inversion* of a musical interval as: "A simple interval with its upper tone transposed an octave downwards. . . Inverted primes become octaves; seconds become sevenths; thirds, sixths, etc."

A mathematical inversion Webster defines as: "A change in the order of terms of a proportion. . ." So what if a fifth is just an inverted fourth? Simplification, that's what! If we divide the chromatic scale right in the middle at the tritone, the bottom half is just the inverse of the upper half. This means that you only need to learn and think about half as much. This is not only true musically

and mathematically, but your own ears will also easily recognize the similarities between an interval and its inverse.

Try the following experiment on any piano or organ that's in tune. Pick out any black or white key and call it 1 for reference. This home note is called the *tonic* and should be located near the center of the keyboard for reasons I'll explain in a moment. Now find note 6 by counting up six keys including 1 and all black and white keys. Now play both 1 and 6 together; that's how a perfect fourth sounds. Try it again with 1 and 8 this time; that's how a perfect fifth sounds. Now go back and forth between 1 and 8 and then 1 and 6 to get a feel for the fifth and its inverse. Next try the same thing with 2 and 7 then 2 and 9. These two intervals are also the fourth and its inverse, the fifth, but you have *transposed* them up by half a step. Now try a minor third and its inverse, the major sixth. First play 1 and 4 together and then 1 and 10 together.

You should notice that the minor third and major sixth don't sound quite as sweet or harmonious as the fourth and fifth did. Now try transposing up a half step to 2 and 5 then another half step to 3 and 6, and so on up the keyboard. Do the same with the fourth, first 2 and 7, then 3 and 8 and so on up the scale. Notice how the fourth and minor third sound similar regardless of the tonic or home key chosen, and how they are clearly different from each other even if played in different octaves; in music as in physics everything is relative to the observer.

You may even want to make a list for yourself of the intervals which sound alike. You can also note which intervals are most harmonious and which are most dissonant, or rough. I'll even bet your list looks like mine! If you think I've biased you, have your friends or family make lists. I'll bet they all are in agreement. Table 4 contains my list, which I've called "Music to Mathematics to Music" for reasons you'll see in a moment.

Interval	Ratio	Inverse	Octave Shift	Musical Inverse
Unison	1/1	1/1	2/1	Octave
Fourth	4/3	3/4	3/2	Fifth
Major Third	5/4	4/5	8/5	Minor Sixth
Minor Third	6/5	5/6	5/3	Major Sixth
Major Second	9/8	8/9	16/9	Minor Seventh
Minor Second*	16/15	15/16	15/8	Major Seventh
Tritone	64/45	45/64	45/32	Tritone

*The minor second is more dissonant to me than the tritone, but the tritone seems more dissonant to me than the major seventh, so the minor second doesn't fit in this list very well.

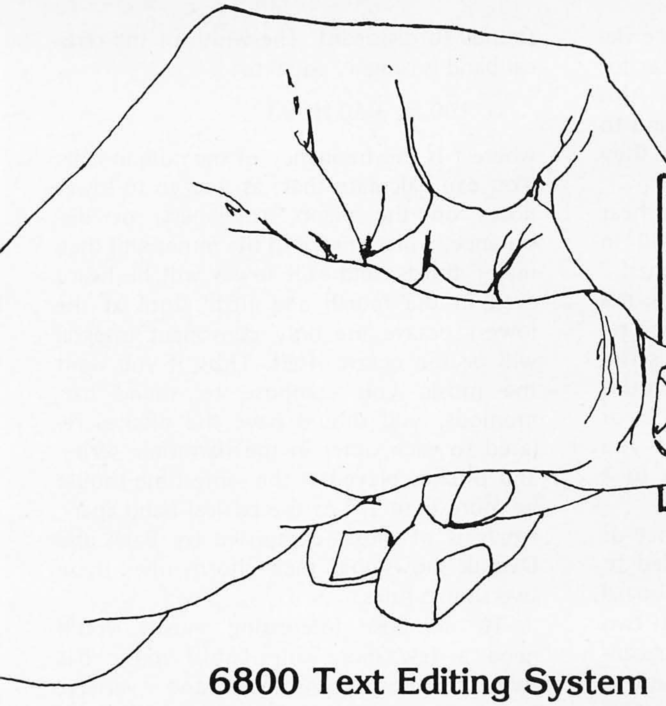
Table 4: Music to mathematics to music. The intervals useful in music are listed in order from the most harmonious to the most dissonant. Most people are in good agreement about the order of evaluation of the relative degrees of harmoniousness in the first five intervals listed.

Now isn't that a remarkable historical achievement: what musicians have been calling an inverse is also an inverse of the frequencies of pitches according to the mathematical definition of inverse. Although I'm neither mathematician nor musician, I have read a number of books on both subjects, including some on the psychophysics of music, and I have never seen this simple and simplifying correspondence of musical and mathematical inverses mentioned. Perhaps it was information lost with the burning of Pythagoras and his temple 2500 years ago. A close look at my list of most harmonious to most dissonant reveals that as the top and bottom of the fractions get larger, the harmony decreases and the dissonance increases, (with the exception of the minor second; but let's forget about this exception for the moment).

The order in this list is no accident; neither is it a learned cultural bias! It is as if we had a brain with a center which continually seeks for simplicity, harmony and order. The *harmonic series:* 1, 2, 3, 4, 5, 6, . . . , is found extensively in man's theories about nature. Is this because it is a property of nature, or is it because man's brain can understand things better if they are in such a series? Such a question is interesting, but can only be raised and not answered in an article about music for computer nuts. Music, like speech, is unique to man and is totally abstract. By abstract, I mean that for the most part, no attempt to copy nature is made.

Music is solely a product of man's brain, or ear-brain combination. Here is where we find harmonic series galore. A musical *chord* such as the *major triad* is three notes played together, the frequencies of the notes being related to each other as elements of a harmonic series are related. In the key of C major, the major triad is C, E, and G which have pitch ratios of 4, 5, and 6 (ie: 4/4, 5/4 and 6/4). Often to make the chord sound fuller, a musician will add the C an octave lower, and the C an octave higher. This also fills out the harmonic series some more: 2, , 4, 5, 6, , and 8. How about the missing 1, 3, 7, 9, etc? You can try 1 and 3 for yourself; they are simply the C an octave lower still, and the fifth up from the next C, and they fit in beautifully.

Unfortunately, you won't be able to try 7 on a piano; it would be 7/4, which is 1.8 percent lower than B$^\flat$, a minor seventh from the C of the triad. Fortunately, if you



build the pure diatonic scale interface described below, you will be able to hear for yourself how well 7 fits into the series. Also you will be able to hear 11, and 13, and to hear how, and under what conditions they fit.

The ear-brain wants so much to hear harmonic series that it will even fill in missing pitches. The "missing fundamental," or the lowest note of a harmonic series, has been studied by many doing acoustics research. If your ear is presented with a series of tones whose frequencies are in the ratios of whole numbers such as 2, 3, 4, 5, 6, or 3, 5, 7, 9, your brain tells you that you actually hear the pitch corresponding to 1 (the fundamental) also!

Now let's get back to the dissonance of the minor second and why you needed to stay in the middle of the piano keyboard to do the experiments with intervals. If two pitches are very close together, the ear cannot tell them apart, and they are heard as a single smooth pitch. If the pitches are far enough apart, two smooth and distinct notes are heard. If the distance between the pitches is in the *critical band*, the two notes are heard as two more or less rough notes. This roughness is maximum at 1/4 of the critical band. It turns out that the minor second is 1/4 of the critical band over the middle of the piano range, and this is why it sounds so dissonant. The width of the critical band is roughly equal to:

$$100 \text{ Hz} + 50 \text{ Hz} \times f$$

where f is the frequency of the note in kHz. You can calculate that, as you go to lower notes on the piano, roughness, or dissonance, will be heard in the minor and then major thirds, and still lower will be heard even in the fourth and fifth, until at the lowest octave the only consonant interval will be the octave itself. Thus, if you want the music you compose to sound harmonious, you should have the pitches related to each other in the harmonic series, and pitches played at the same time should be more than 1/2 of the critical band apart. Analysis of music composed by Bach and Dvorak shows that their chords obey these two simple rules.

To compose interesting music, you'll need a few more rules. Most music has two features, constancy and variety. It is as if the brain center which looked for order, simplicity, and harmony was easily bored, so once it found a pattern, it would soon be looking for another. Our musical needs vary. Sometimes we want very simple tunes so we can unwind, and at other times we need complex melodies to keep our interest. Once you have made a tune with a computer, it will be possible, in principle,

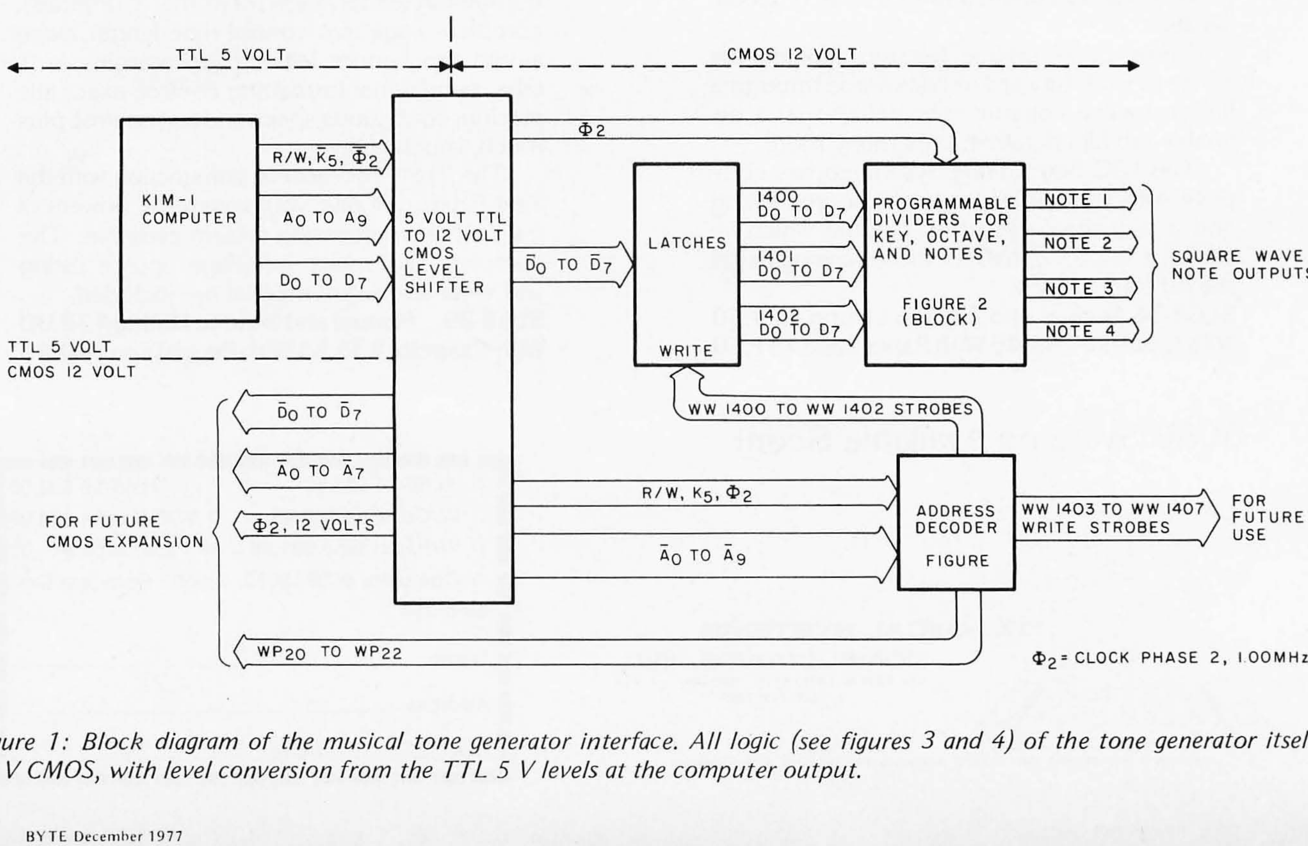


Figure 1: Block diagram of the musical tone generator interface. All logic (see figures 3 and 4) of the tone generator itself is 12 V CMOS, with level conversion from the TTL 5 V levels at the computer output.

for you to have the computer make all sorts of variations of your tune to keep it interesting.

Now let's switch gears from music to computer oriented electronics and find out how to build a diatonic computer music interface. The interface costs less than $19 to build, including 24 integrated circuits, LEDs, resistors, four diodes, and a universal type printed circuit board. It will put out four notes simultaneously, and will play in almost nine octaves (17.36 Hz to 7812.5 Hz). The highest and lowest octaves have 12 and 13 different notes, and the middle seven octaves have 33 notes each, giving a total of 256 unique notes. It uses three bytes of memory space.

The interface can be functionally divided (see figure 1) into four parts:

- A set of programmable frequency dividers.
- A three byte latch.
- An address decoder.
- A level shifter to change the 5 V signals from the computer to 12 V signals for the CMOS circuits so they can operate fast enough to follow the 500 ns write pulse put out by the computer.

The block diagram shows that I've chosen hexadecimal addresses 1400 to 1402 to drive the interface. This is a convenient memory location for me because I have a KIM-1 with 12 K of memory and these locations are not used for anything else. You'll notice that I've also decoded write pulses for hexadecimal addresses 1403 to 1407 and page selects of addresses 20XX to 22XX for future expansion of the interface. Also, eight address lines, eight data lines, a clock line, and the write pulses for addresses 20XX to 22XX, all at 12 V, are brought out to the edge connector for use with other CMOS interfaces.

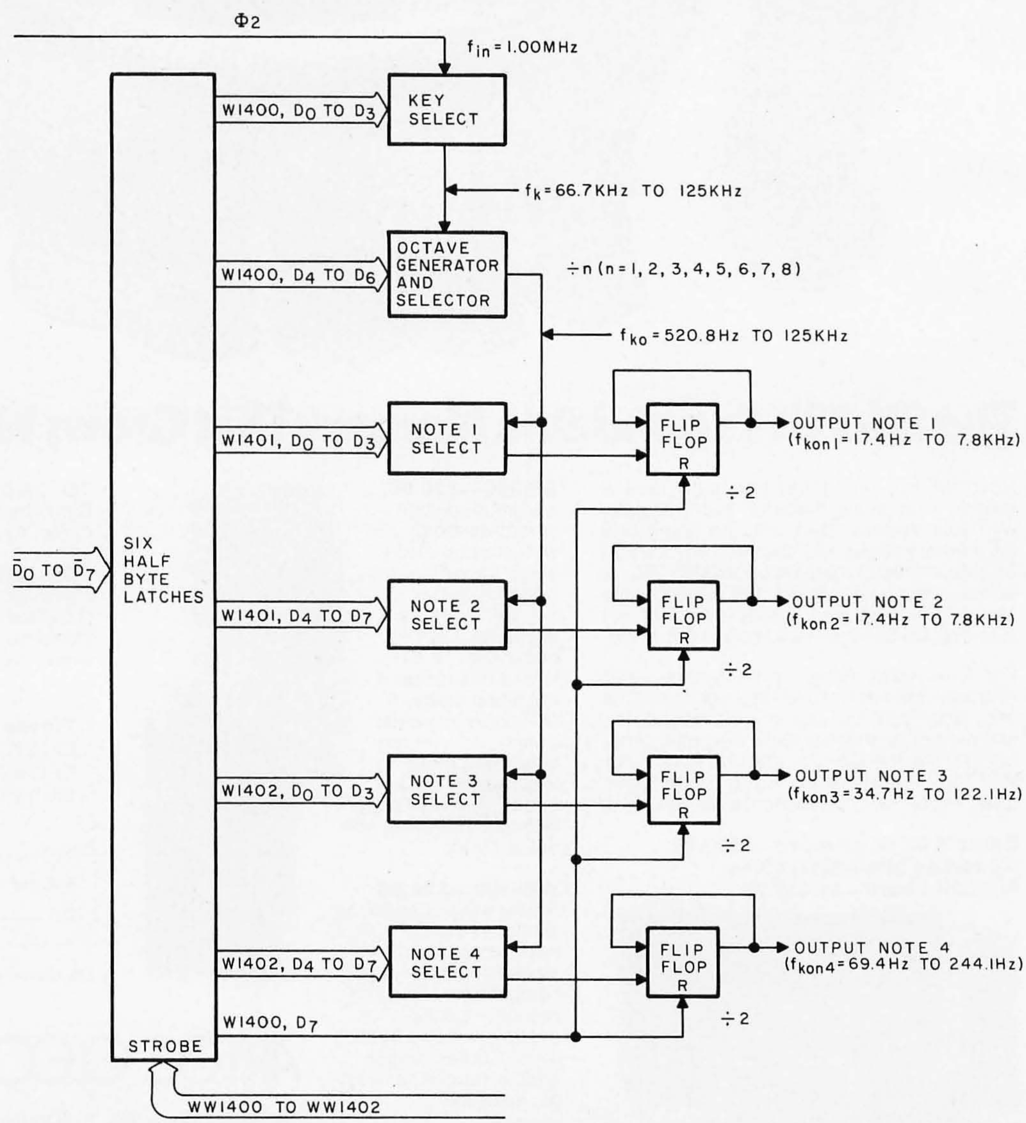
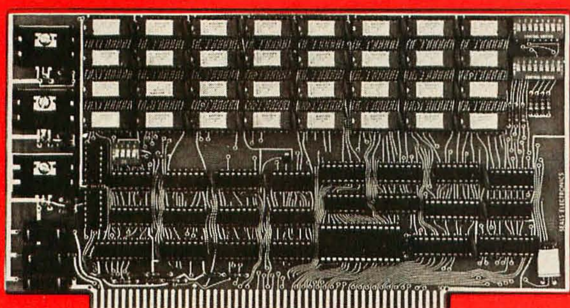


Figure 2: Detail block diagram of the tone generator, which uses the 1 MHz clock of the KIM-1 as its frequency standard. The outputs at right are square wave signals which can be sent to further filtering and signal processing before mixing down to one or two stereo channels.

SEALS SELLS IT ALL

All Boards Assembled and Tested

DYNAMIC RAMS

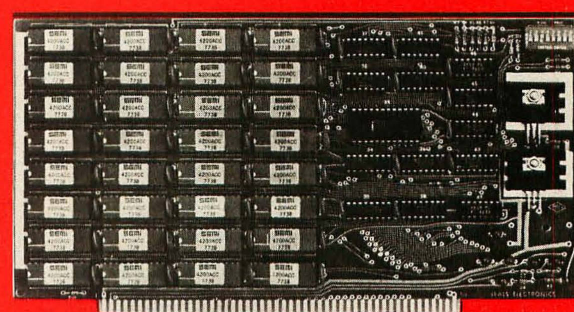


16 KDS • 32 KDS • 64 KDS

- Bank Selectable via Software
- Addressable in 4K increments
- On Board Refresh
- ROM phantom line to inhibit RAM for ROM
- Addressing Switch Selectable
- Solder Mask Both Sides
- Top Quality FR-4 fire resistant glass board material

16KDS **$395.00**
32KDS **750.00**
64KDS **1,850.00**

STATIC RAMS



8 KSC-X • 8 KSC-Z • 16 KSC-A

- Access times to 250 nsec.
- Address selectable by DIP Switch
- All Address, Control, and Data outlines fully buffered
- Low Power Consumption
- Meets all DMA requirements
- Solder Mask both sides of P.C. boards
- Top Quality FR-4 fire resistant glass board material

8KSC-X (500 nsec.) **$229.00**
8KSC-Z (250 nsec) **269.00**
16KSC-A (250 nsec.) **579.00**

ALL BOARDS S-100 BUS COMPATIBLE

PERIPHERAL UNIVERSAL PROCESSOR - PUP-1

Have you seen the Seals Electronics, Inc., Peripheral Universal Processor-PUP-1? It is the newest member of the Seals family of performance products. The PUP-1 is designed to meet all the needs and demands placed on a continuous-duty micro-computer by OEM applications. Write or call Seals factory direct for more information.

ORDER DELIVERY: FROM STOCK TO 30 DAYS

B-12-77

SEALS ELECTRONICS, INC.

10728 DUTCHTOWN RD. ▪ CONCORD, TN. 37720

PHONE 615-966-8771 **TELEX NO. 55-7444**

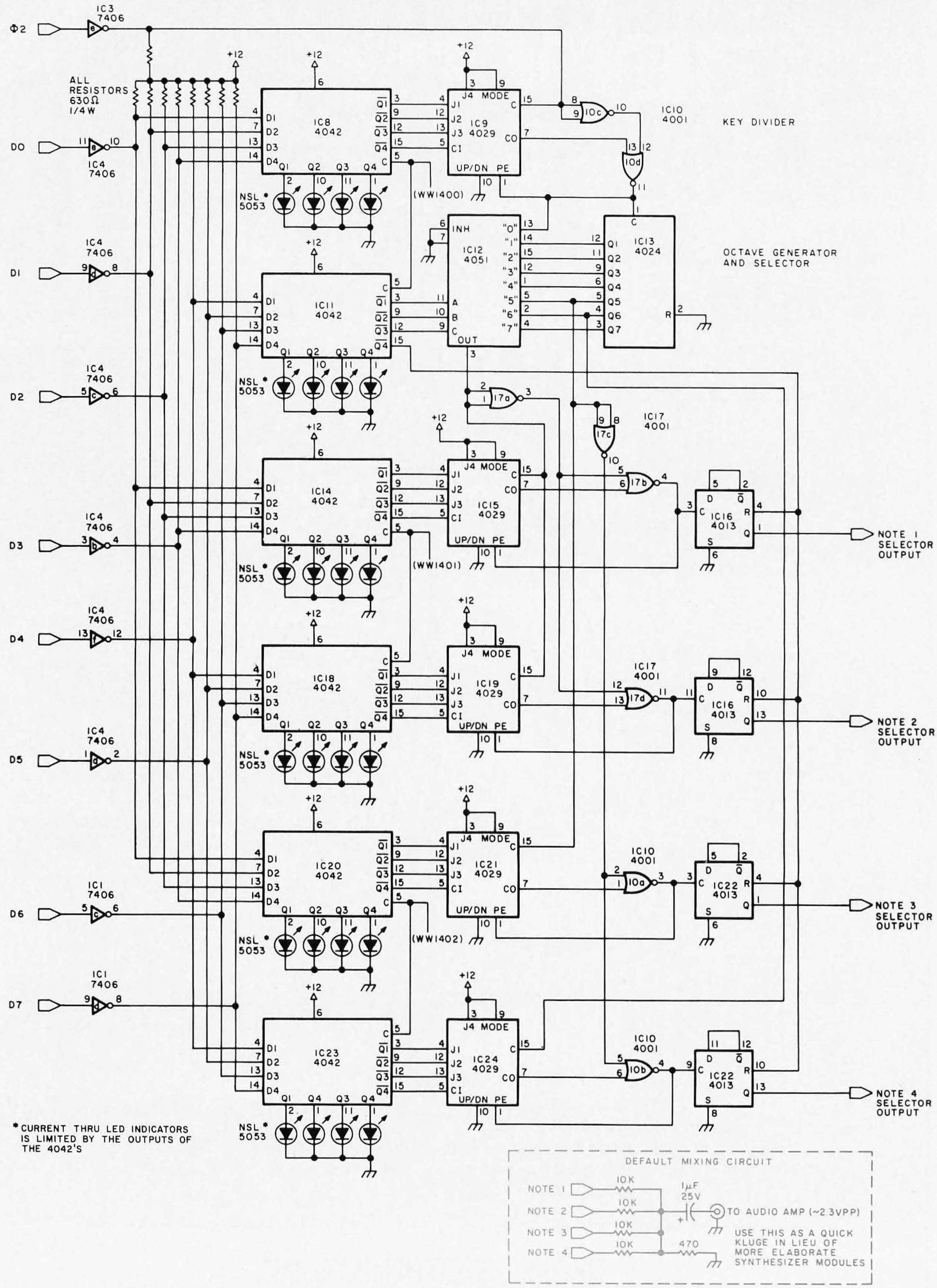
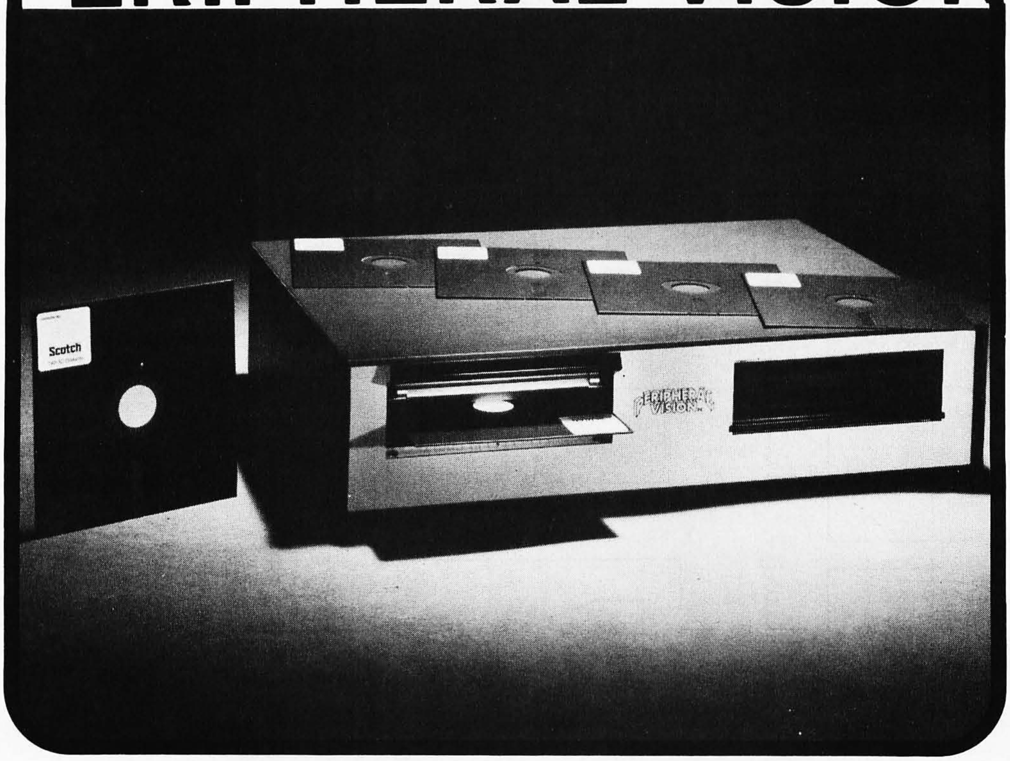


Figure 3: Schematic of the tone generator's key, octave and note selection logic. A default mixing circuit is shown to allow connection of all four outputs directly to one audio amplifier for testing.

take a close look at
PERIPHERAL VISION



Peripheral Vision is a young, fast-moving company that's dedicated to selling reasonably priced peripherals for various manufacturers' CPU's.

So now, when you build your microcomputer system, you'll know where to look for all the peripherals that will make your system do what it's supposed to do.

Peripheral Vision may be young, but we have some old-fashioned ideas about how to run our business.

We know there are serious incompatibilities among the various manufacturers' peripherals and CPU's. We want to get them together. And we want to bring significant new products to market--products consisting of everything from adaptation instructions/kits for hardware and software to major new designs.

Most important to our customers, Peripheral Vision is committed to helping you get along with your computer. We'll do all we can to make it easy.

Our first product is a real reflection of this philosophy. It's a full-size floppy disk for the Altair-Imsai plug-in compatible S-100 BUS. And it's available for as low as $750.00.

Our floppy disk has many exciting features:

- 1 interface card supports 4 or more drives
- Stores over 300,000 bytes per floppy
- Bootstrap EPROM included--no more toggling or paper tape
- Completely S-100 plug-in compatible
- Drive is from Innovex (the originator of the floppy concept)--assembled and tested
- Disk operating system with file management system included on floppy
- Cabinet and power supply optional

Also in the works are many new products we'll be letting you know about soon, if you'd like to take a closer look. Like I/O cards, tape drives, an impact printer--all for the S-100 BUS--and we're designing peripherals for a lot of other CPU's too.

We've given you a little glimpse of who we are and what we're doing. If you want to see more, just fill in the coupon below.

PERIPHERAL VISION INC.

P.O. Box 6267/Denver, Colorado 80206 303/777-4292

Name ____________________

Address ____________________

City/State/Zip ____________________

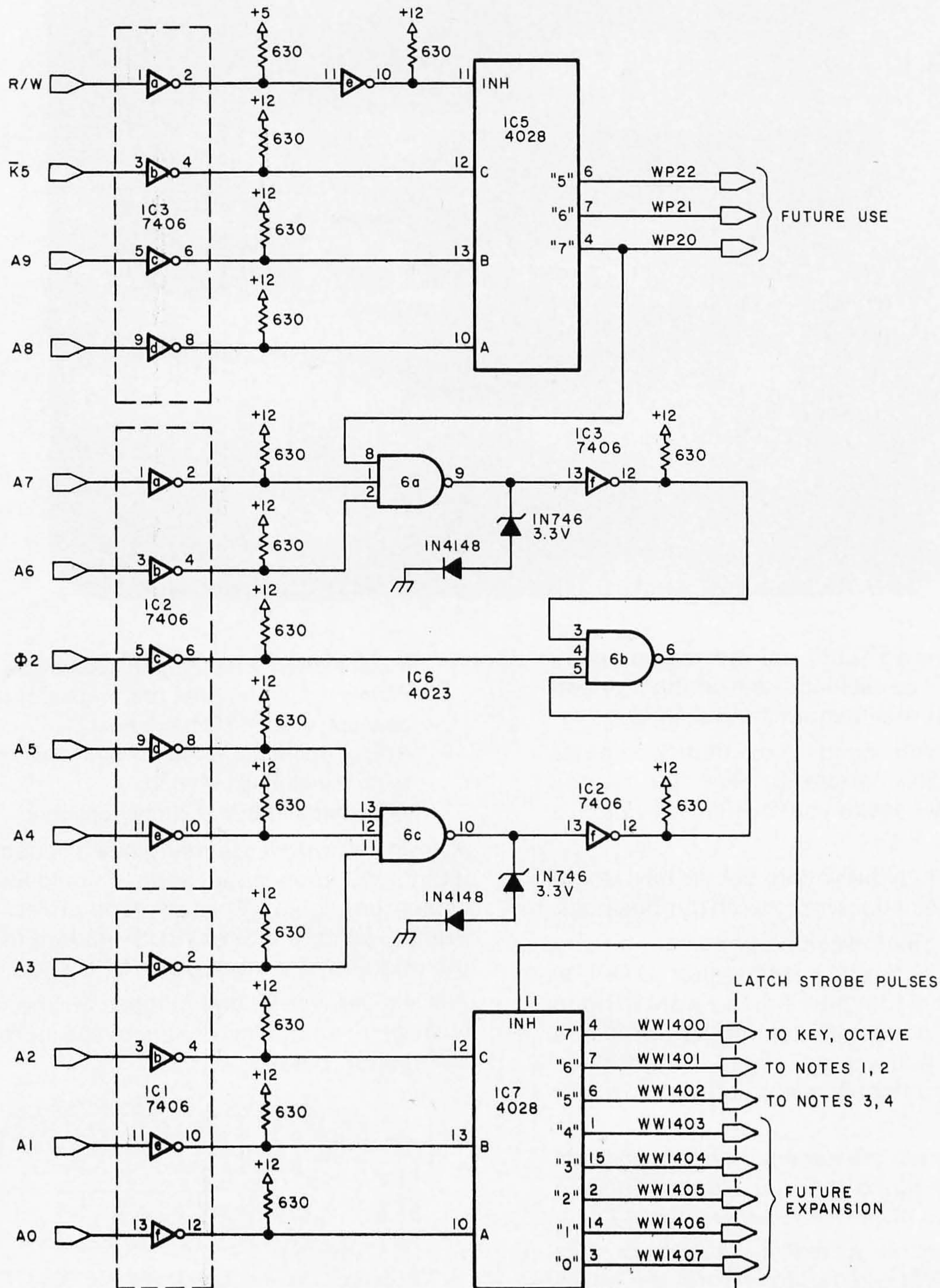


Figure 4: Schematic of the tone generator's KIM-1 address space decoding, a diagram of the edge connector, and power wiring table for figures 3 and 4.

The six programmable dividers are the heart of the interface (see the detail block diagram of figure 2 and circuit diagram of figure 3). Five of these are 4029 presettable, bidirectional, binary or decade counters set up to count down in binary mode. In this mode the carry out (CO) line goes low whenever the counter counts down to 0. The CO signal is inverted and returned to the preset enable (PE) input which sets the counter to the value of the binary number on input pins J_4 to J_1. Each positive transition of the clock (C) input causes the counter to count down by one as long as the clock inhibit (CI) is low. Because J_4 (pin 3 on IC9, IC15, IC19, IC21, and IC24) is always high, the counters may be set to divide by 8, 9, 10, 11, 12, 13, 14, or 15, depending on the binary number on inputs J_3 to J_1. This number is stored in 4042 latches by writing the data into D_3 to D_1 of the latch as if it were a memory location. For example, if a binary three (011) were on J_3 to J_1, 8+3, or 11 would be loaded into the counter when PE went high, and the C input would have 11 positive transitions before CO would go low, forcing PE high momentarily, and again loading the counter with 11. Thus the frequency of PE pulses would be 1/11 of the frequency of positive transitions at C. A flip flop at the output of the note dividers converts the PE impulses into square waves with a 50 percent duty cycle. Each of the Q_4 latch outputs turns off a divider and thus turns off the sound of one or more note outputs. Bits 4 and 8 of hexadecimal location 1400 turn off all the sound, whereas bits 4 and 8 of address 1401 and 1402 turn off notes 1 thru 4, respectively. The reason for all this turn off is that music has a lot more silence in it than is generally recognized. To make notes sound distinct, rather than all run together, the sound must be shut off for periods of 10 to 50 ms (for example).

The key selector divides the computer's 1 MHz clock by a number from 15 to 8 to produce frequency f_K of 66.7 kHz to 125 kHz as shown in the second block diagram. A binary divider, IC13, produces seven more octaves (factors of two in frequency) from f_K, and the 1 of 8 selector, IC12, selects one of the octaves, f_{KO} (520.8 Hz to 125 kHz), based on bits 6 to 4 stored at

Power Wiring Table

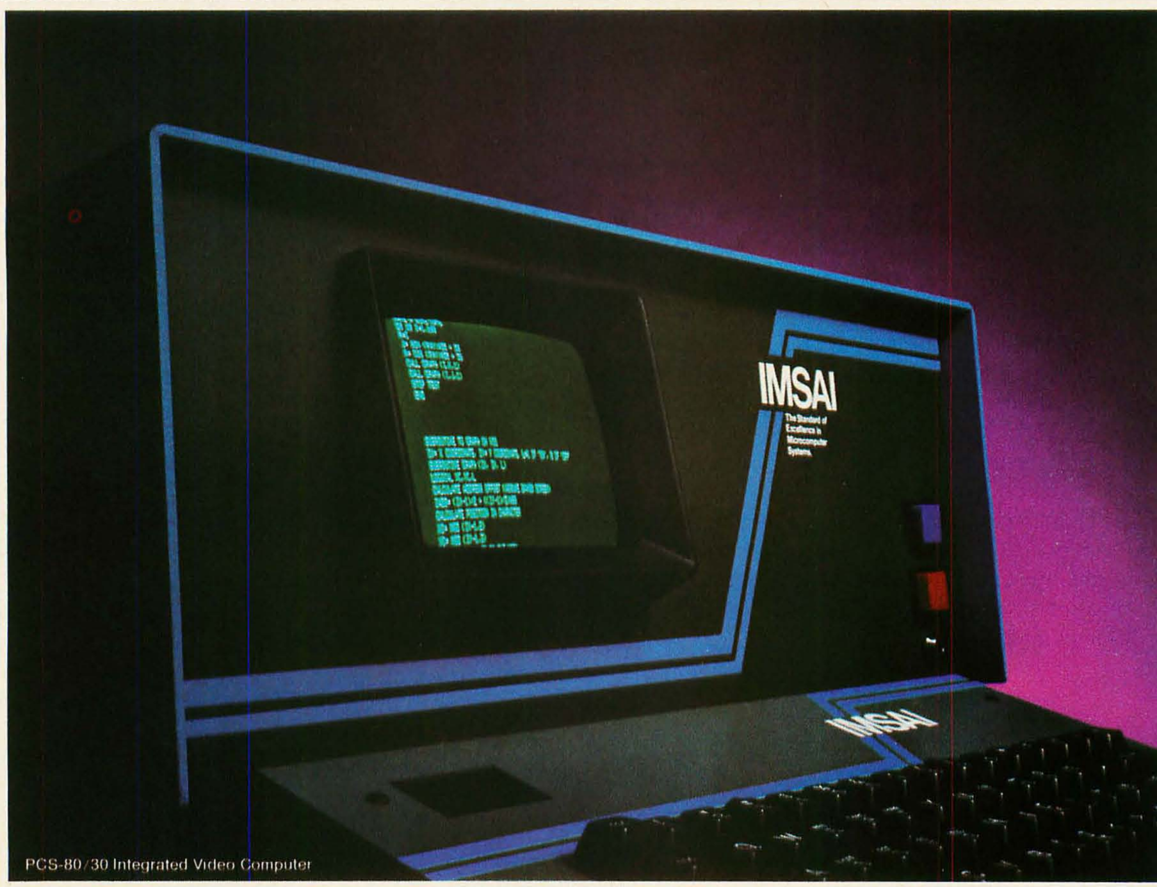
Number	Type	+5 V	GND	+12 V
IC1	7406	14	7	–
IC2	7406	14	7	–
IC3	7406	14	7	–
IC4	7406	14	7	–
IC5	4028	–	8	16
IC6	4023	–	7	14
IC7	4028	–	8	16
IC8	4042	–	8	16
IC9	4029	–	8	16
IC10	4001	–	7	14
IC11	4042	–	8	16
IC12	4051	–	8	16
IC13	4024	–	7	14
IC14	4042	–	8	16
IC15	4029	–	8	16
IC16	4013	–	7	14
IC17	4001	–	7	14
IC18	4042	–	8	16
IC19	4029	–	8	16
IC20	4042	–	8	16
IC21	4029	–	8	16
IC22	4013	–	7	14
IC23	4042	–	8	16
IC24	4029	–	8	16

Edge Connector Wiring Diagram

Group	Signal	Pin	Pin	Signal	Group
	+5 V	1	A	ground	
	12 V output ϕ_2	2	B	write page 20	12 V outputs
5 V inputs	$\overline{\phi}_2$	3	C	write page 21	12 V outputs
5 V inputs	R/W	4	D	write page 22	12 V outputs
5 V inputs	$\overline{K}_5$	5	E	$\overline{D}_0$	12 V outputs
5 V inputs	D_0	6	F	$\overline{D}_1$	12 V outputs
5 V inputs	D_1	7	H	$\overline{D}_2$	12 V outputs
5 V inputs	D_2	8	J	$\overline{D}_3$	12 V outputs
5 V inputs	D_3	9	K	$\overline{D}_4$	12 V outputs
5 V inputs	D_4	10	L	$\overline{D}_5$	12 V outputs
5 V inputs	D_5	11	M	$\overline{D}_6$	12 V outputs
5 V inputs	D_6	12	N	$\overline{D}_7$	12 V outputs
5 V inputs	D_7	13	P	$\overline{A}_0$	12 V outputs
5 V inputs	A_0	14	R	$\overline{A}_1$	12 V outputs
5 V inputs	A_1	15	S	$\overline{A}_2$	12 V outputs
5 V inputs	A_2	16	T	$\overline{A}_3$	12 V outputs
5 V inputs	A_3	17	U	$\overline{A}_4$	12 V outputs
5 V inputs	A_4	18	V	$\overline{A}_5$	12 V outputs
5 V inputs	A_5	19	W	$\overline{A}_6$	12 V outputs
5 V inputs	A_6	20	X	$\overline{A}_7$	12 V outputs
5 V inputs	A_7	21	Y	A_9 5 V input	
5 V inputs	A_8	22	Z	+12 V	

Continued on page 170

the best way to build your system...



PCS-80/30 Integrated Video Computer



PCS-80/15 (w/IKB-1)



PCS-80/30; 80-25A; DHMO



PCS-80/ Components

IMSAI introduces the PCS-80 component system...

is to take it apart piece by piece.

Compare the features of our S-100 bus system, the industry standard, to anyone else. After you've taken them apart piece by piece, you'll know why IMSAI is the system you can grow with. At a price you can live with.

IMSAI 80/30 Integrated Video Computer (with Intelligent Keyboard—IKB-1) Standard Features:

□ **Price assembled $1499.** IMSAI is the only S-100 bus manufacturer that offers a microprocessor driven keyboard with "N" key roll over, 2¼K of RAM, 8 expansion slots, choice of 4K, 16K, 32K and 64K RAM expansion boards, 3K ROM monitor, synch/asynch serial interfaces, parallel and serial ports, high resolution CRT monitor, 24 x 80 display with graphic editing and data entry features, and 28 amp power supply for the incredibly low price of $1499.

□ **mpu Speed.** IMSAI is the only S-100 bus manufacturer that offers true 8080 compatibility, operating at 3 mHz.

□ **RAM Included.** 2¼K.

□ **Expansion Slots.** Eight expansion slots are provided in a new terminated and regulated motherboard (10 slots total).

□ **RAM Board Sizes.** IMSAI is the only S-100 bus manufacturer to supply 4K, 16K, 32K, and 64K RAM memory expansion boards.

□ **ROM Monitor.** IMSAI is the only S-100 bus manufacturer to provide 3K of ROM.

□ **Asynch/Synch.** Only one other S-100 bus manufacturer provides both methods of data communication.

□ **PIO/SIO.** IMSAI is the only S-100 bus manufacturer that provides two serial ports and one fully implemented parallel port at no extra charge.

□ **Video I/O.** IMSAI is the only S-100 bus manufacturer to include a high resolution (14 mHz) monitor as an integrated part of the computer.

□ **CRT Format.** IMSAI is the only S-100 bus manufacturer to provide a full 24 x 80 screen, which is two times the capacity of the common 16 x 64 screen.

□ **Graphic/Edit.** IMSAI is the only S-100 bus manufacturer that provides graphics and text editing features with character and line insert/delete for your CRT display.

□ **Keyboard Included.** IMSAI is the only S-100 bus manufacturer to supply a microprocessor driven keyboard with "N" key roll over and tiered construction for a true typewriter keyboard touch.

□ **28 amp Power Supply.** The world famous IMSAI power supply assures stability and reliability of performance.

Options: IMSAI is the only S-100 bus manufacturer to provide a comprehensive array of fully integrated options including: line and character printers, CRT terminals, intelligent keyboard, ACR storage, standard and mini floppies, TTY BASIC with OS, 4K, 8K and 12K BASIC, audio cassette BASIC with OS, 8K disk operating system (DOS) based upon CP/M,* scientifically and commercially oriented disc BASIC and level 2 FORTRAN IV compiler.

□ **Printers.** Only one other S-100 bus manufacturer can supply both line and character printers.

□ **CRT/Keyboard.** IMSAI is the only S-100 bus manufacturer to provide both CRT terminal and intelligent keyboard as separate options.

□ **ACR Storage.** Available.

□ **Floppies.** IMSAI is one of the few S-100 bus manufacturers to provide both standard and mini floppies and the only S-100 bus manufacturer that supplies double density standard floppies.

□ **TTY BASIC.** IMSAI is one of the few S-100 bus manufacturers that provides self-contained operating systems with 4K, 8K and 12K BASIC.

□ **ACR BASIC.** IMSAI supports ACR BASIC with an 8K version.

□ **DOS.** IMSAI is the only S-100 bus manufacturer to provide an enhanced version of the control program monitor (CP/M*) that can support up to 18 disk drives.

□ **Disc BASIC.** IMSAI is the only S-100 bus manufacturer that provides both scientific and commercial versions of compiler oriented BASIC.

□ **FORTRAN IV.** IMSAI is the only S-100 bus manufacturer that offers a level 2 FORTRAN IV compiler that operates under an enhanced version of CP/M.*

Prices and specifications subject to change without notice.
*CP/M is a trademark of Digital Research Corporation.

Price/Performance no one else has put together.

IMSAI®

The Standard of Excellence in Microcomputer Systems

IMSAI Manufacturing Corporation, 14860 Wicks Blvd., San Leandro, CA 94577 (415) 483-2093 TWX 910-366-7287

Circle 61 on inquiry card.

Technical Forum

The XF and X7 Instructions of the MOS Technology 6502

H T Gordon
Dept of Entomological Sciences
110 Wellman Hall
University of California
Berkeley CA 94720

None of the operation codes listed in the instruction set of the MOS Technology 6502 has a low order bit pattern of 1111 (hexadecimal F) or 0111 (hexadecimal 7), although many have 1110, 1101, 0110 or 0101. How does the processor interpret op codes of type XF or X7 where X is an arbitrary high order digit?

When I tried this out I found that it executes them as valid instructions to do *both* the XE and XD, or *both* the X6 and X5 instructions, with fascinating and useful results. Thus for example A5 XX causes a load of the byte in page zero location XX into the accumulator, A6 XX loads this byte into the X register, and A7 XX loads it into *both* index registers, an operation that would need three program bytes with the conventional coding.

Op code 87 is even more interesting, since 85 XX stores the value in the accumulator into page zero location XX and 86 stores the value in the X register into XX, and it is obviously impossible to store two different values simultaneously in one location. The effect of 87 XX is to store into location XX only those 1 bits that occur in *both* the accumulator and the X register. The effect is that of a logical AND between the accumulator and the X register (neither one being altered), with storage of the result into XX.

The effect of 97 XX resembles that of 87 XX, except that the result of the AND is stored in page zero location XX + Y, since Y indexed storage is used by op codes 95 and 96. Since these are store instructions, no testable flags are set. In general, op codes of type XD and XE have the same logic as X5 and X6, differing only in having a 2 byte absolute address instead of a 1 byte page zero address. This is not true for op codes of type 9X. Both are nevertheless executed, but the operations are not the same as those of 96 and 97. They resemble the valid 9D in that storage into a Y indexed absolute address (XXXX + Y) is commanded. However, 9E stores the result of an AND between the byte in the X register and hexadecimal immediate value 02, ie: if bit 1 of this byte is a 1, 02 is stored, and if not, 00 is stored. (I have no idea where the processor finds the 02.) Op code 9F stores the result of an AND between the value 02, the X register, and the accumulator; if the bytes in *both* registers have a 1 in bit 1, 02 is stored; otherwise 00 is stored.

When X6 and X5 command different operations, X7 causes the X6 to be done first, followed by the X5. For example E7 XX causes the byte in XX to be incremented by 1, then subtracts this value from the accumulator, setting the proper flags and leaving the result in the accumulator. Everything the "new" instructions do could of course also be done with the conventional set, using more program bytes and time. Most of the unused op codes of the 6502 "run," but 12 of type X2 cause operations to become "lost in space," from which only RESET can rescue them. It would be interesting to know whether some of the unused op codes of other microprocessor designs will also prove to be valid instructions.

Since discovering the XF and X7 instructions, I have found that there are also "unofficial" XB instructions. For example A9 XX is a load immediate of the byte XX into the accumulator, and AA commands transfer of the byte in the accumulator into the X register. The effect of AB XX is to load XX into *both* the accumulator and X registers, setting the usual flags. Some other XB op codes are also executed, but I have not yet had time to work out their operation. I have no doubt that some of these instructions could be put to use, but there is a hazard. If manufacturers decide to add new *planned* instructions in chip redesign, programs using the unplanned ones will be incompatible.■

Technical Forum is a feature intended as an interactive dialog on the technology of personal computing. The subject matter is open-ended, and the intent is to foster discussion and communication among readers of BYTE. We ask that all correspondents supply their full names and addresses to be printed with their commentaries. We also ask that correspondents supply their telephone numbers, which will be printed unless we are explicitly asked to omit them.

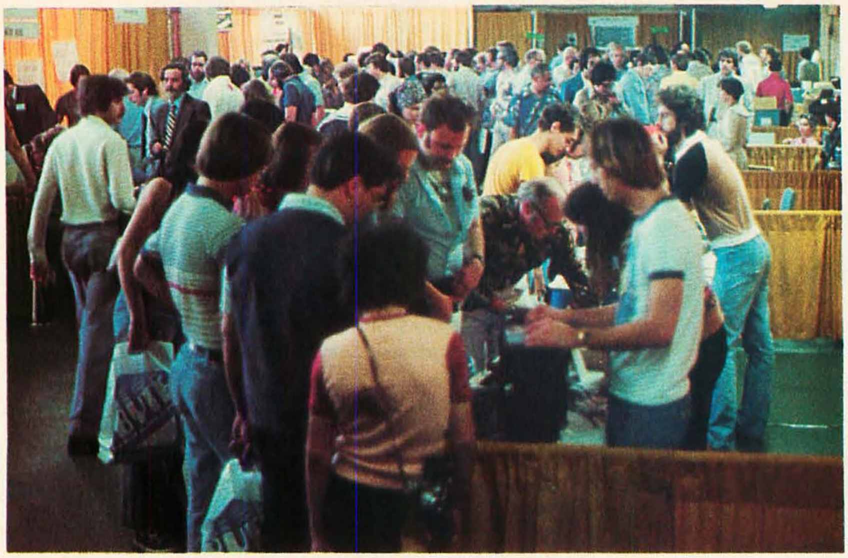


Photo 1: A view of the PC 77 convention floor.

PC 77

By Chris Morgan, Editor



Photo 2: Heath's display, a popular spot at the convention where hackers could get their hands on the new Heathkit computers for the first time.

Who would go to Atlantic City *before* the gambling casinos are built? Computer freaks, that's who! Thousands of microcomputer enthusiasts (between 8 and 12 thousand) crammed into the Shelburne hotel August 27 and 28 to see over 150 manufacturers display their computer creations. Exotic sights met the eye at every turn: a complete mock up of a fighter plane was being run by a microcomputer, with plans available to intrepid experimenters who wanted to duplicate the gadget; games galore and some very high class color graphics filled video screens; computers spoke and listened to fascinated onlookers; Heath, Commodore and Radio Shack displayed their brand new microcomputer systems; and BYTE was on hand at a brand new convention booth.

Everything went smoothly from first hour to last, and the well attended banquet featured a host of speakers including computer pioneer Dr John Mauchly.

Next year's convention promises to be even bigger and better at its new location: the Atlantic City Convention Center.■



Photos by Charles Floto

Photo 3: One of the many color video displays on view at Personal Computing 77. Shown are two Compucolor displays offered by the Computer Mart of NJ.

Photo 4: The Computer Store (New York area) booth.

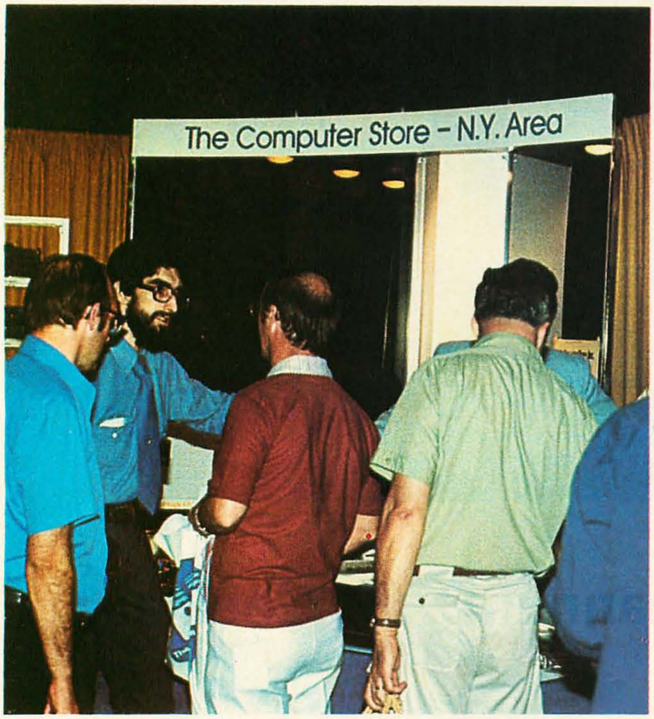




Photo 5: One corner of the expansive MITS booth which took up over 800 square feet and featured the complete line of Altair computers.

Ciarcia's Circuit Cellar

On a Test Equipment Diet?

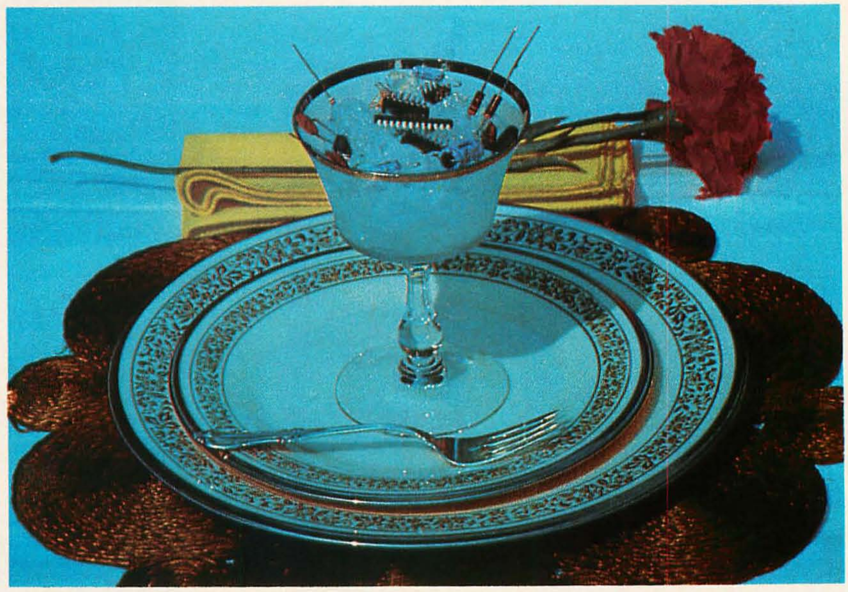


Photo 1: ICs on ICE. Pictured are all the components necessary to build the 8 channel 3½ digit computer controlled voltmeter described in this article.

Try an 8 Channel DVM Cocktail!

Steve Ciarcia
POB 582
Glastonbury CT 06033

About three weeks ago, I was testing a new 8 bit analog to digital converter which I had just built for an upcoming magazine article: this one, in fact. It was a high speed successive approximation analog to digital converter which performed 200,000 conversions a second, and it worked fine. I had intended to use it for some speech digitization experiments. During the testing phase, however, I became exasperated from continually moving my digital voltmeter (DVM) probes around the circuit to take readings and having to stop to make the same calculations repeatedly. To speed the process up, I wrote a BASIC program which would do the number crunching, provided I typed in the voltage values correctly. More often, though, all I wanted was to monitor a few voltage levels simultaneously.

After stringing my two DVMs, an analog volt-ohm meter (VOM), and my oscilloscope all over the bench to aid in my testing, I concluded that there must be a better way. It's old hat to use one channel of a dual trace scope to troubleshoot the other trace, so it was natural to consider using the analog to digital converter to monitor itself. While the thought was momentarily gratifying, the low resolution inherent with eight bits and clumsy binary conversion made me reconsider.

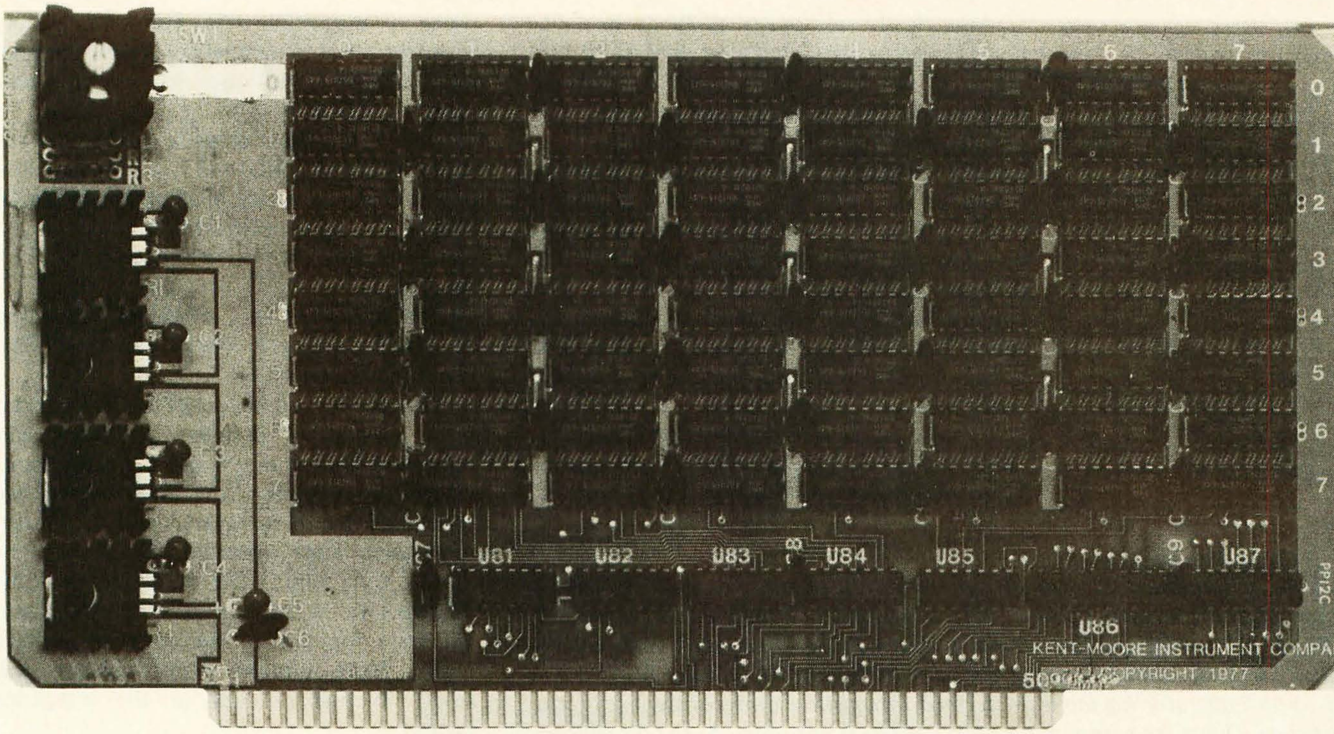
While thinking over this dilemma I was leaning back in my reclining desk chair with one elbow on my computer and my feet up on my printer. I realized that I should move some of the junk so that I'd have more room in the basement. I concluded that what I needed were eight DVMs. This insane desire was quickly eradicated and replaced by a more economically sound idea. I had designed a 4 channel 8 bit digital to analog converter to run with BASIC. It was only natural to design a multichannel analog to digital converter which also interfaced to BASIC.

12 bit analog to digital converters and 3½ digit DVM chips come in a variety of configurations. Converters which specifically state that they are 12 bit converter modules can have either binary or binary coded decimal (BCD) outputs, but are almost universally parallel binary output devices. The end of conversion signal results in immediate data output. The computer just has to scan the data lines and translate them into meaningful notation. Chips which

Motorola MC14433 3½ Digit Analog to Digital Converter Specifications:

- Accuracy: ± 0.05% of reading ± 1 count
- Two voltage ranges: 1.999 V and 199.9 mV
- Up to 25 conversions per second
- Input impedance > 1000 megohms
- Auto zero
- Single positive voltage reference
- Auto polarity
- Drives CMOS or low power Schottky loads
- On chip system clock
- Over, under, and auto ranging signals available

MORE BYTES FOR YOUR BUCK !



8K RAM...assembled, burnt-in, tested and warranted...only 197^{50}!

These are not kits, but completely assembled and tested boards, with a 1 year warranty. No soldering, no messing, no chance of mis-connections . . . just plug 'em in and you're ready to run. The 8K RAM has the same features and speed as our 4K RAM (500 nsec, no wait states) and if that isn't fast enough there's our 8K (Z) fast RAM that gives you a 250 nsec access time for only $217.50!

Because of the low power memory chips used, power requirements are lower than many other RAM boards. And the 8K RAM uses less power than two 4K boards. All RAMs are manufactured to military specification MIL STD-883-C, assuring greatest control over reliability.

Address selection is easily accomplished by our Visaddress™, an easy to read switch on the board top. The 8K board is designed to be selected as one of eight possible 8K RAM boards present on the S-100 bus.

To achieve address selection, the top address lines are decoded using the Visaddress switch. The switch will then show the selected starting address of the RAM card. (i.e. Ø=ØØØØ–1FFF, 2=2ØØØ-3FFF, etc. on the 8K board).

Both boards have fully buffered address and data lines, and extensive built-in noise immunity circuitry. And are plug-in compatible with the S-100 bus (Altair 8800, IMSA1 8080, etc.)

Quality, assembled boards at less than kit prices. But what else should you expect from a company whose prime products are electronic test instrumentation and microprocessing components?

Also available: 4K RAM; $107.00, Alpha-VDM; $107.00, Alpha-VDM-II; $145.00, Graphics-VDM; $137.00.

Order direct by check, BankAmericard or Master Charge (Add $1.50 shipping, credit customers give us all the card numbers, please and Ohio residents add 4½% sales tax) or contact us for more information. Kent-Moore Instrument Company, a subsidiary of Kent-Moore Corporation (founded in 1919), P.O. Box 507, Industrial Ave., Pioneer, Ohio 43554. (419) 737-2352. Or, Kent-Moore of Canada, 246 S. Cawthra Rd., Mississauga, Ontario L-A3P2, Canada.

Kent-Moore
INSTRUMENT COMPANY

Circle 68 on inquiry card.

are specifically referred to as 3½ or 4½ digit DVM large scale integrated circuit (LSI) chips do not have this luxury. In general, their output is a combination of serial and parallel, one digit at a time. Interfacing to a parallel output analog to digital converter would be far easier with regard to the computer software, but as is generally the case, one never gets something for nothing. 12 bit parallel analog to digital converters are expensive. Most are designed to cover high speed data acquisition applications. Speed (1000 to 100 K conversions per second) costs money.

This leaves us with the 3½ digit DVM LSI chips. They run very slowly by comparison (1 to 50 conversions a second), but cost an order of magnitude less. Software to perform the serial to parallel conversions is a bit more involved, but once it's written, who cares?

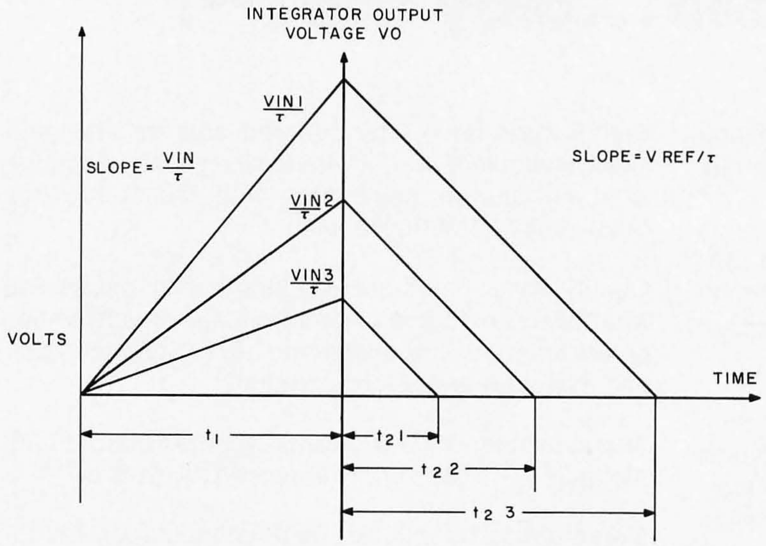
One of the latest chips to hit the market is the Motorola MC14433, a 3½ digit low power complementary MOS analog to digital converter. Its specifications (relative to computer applications) are listed in the box on page 76.

The MC14433 is a modified dual ramp integrating analog to digital converter. This is outlined in figure 1.

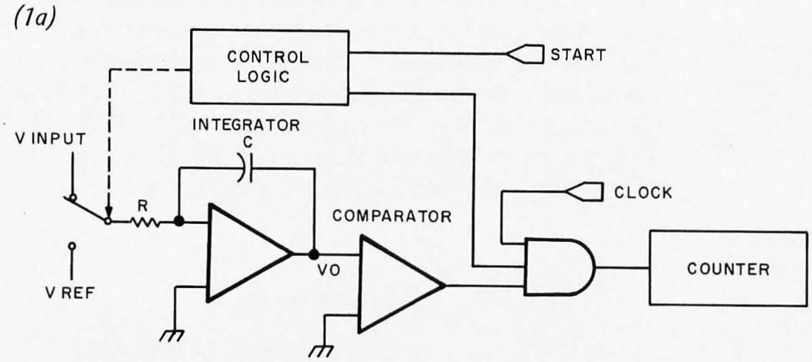
The conversion sequence is divided into two integration periods: unknown and reference. During the V_{in} or unknown input integration sequence, the unknown voltage is applied to an integrator with a defined integration time constant for a predetermined time limit. The result is that the voltage level at the output of the integrator will be a function of the unknown voltage input. More positive input voltages will result in higher levels at the integrator output.

During the second cycle of the integration sequence, V_{in} is replaced at the input of the integrator with a negative 2.000 V reference. The output of the integrator starts to move toward zero while the digital circuitry in the chip keeps track of the time it takes to make it to zero again. The time difference between the two integration sequences is then a function of their voltage difference. Since the integration time constants are the same for both periods, if 2.000 V were the unknown applied voltage, t_2 would be equal to t_1. The unknown voltage is equivalent to the ratio of the periods, times the voltage reference, V_{ref}. This is also known as a ratiometric converter. Quite a mouthful. The full scale range of the converter is determined by the level of V_{ref}. Changing V_{ref} to .200 V will make the same 1999 count represent a 199.9 mV full scale. (Obviously, V_{ref} could be set to any value within the voltage limitations of the chip. But, remember, full scale will still be

Figure 1: A simplified functional representation of the Motorola MC14433 3½ digit analog to digital converter. 1a shows a block diagram of the device; 1b shows the two integration periods used to convert the input voltage to a 3½ digit decimal number. During time t_1, the unknown voltage (V_{in}) is applied to an integrator having a predefined integration time constant (τ) for a preset time. During t_2 a known negative voltage is presented to the integrator. The time needed for the integrator to return to the 0 level is therefore a function of the unknown voltage. A digital counter keeps track of this time, from which V_{in} can be calculated.



(1a)



τ = integration time constant
t_1 = unknown voltage integration period (constant)
t_2 = reference voltage integration period (variable)

$$V_0 = \frac{V_{in} t_1}{\tau} = \frac{V_{ref} t_2}{\tau}$$

that is

$$\frac{V_{in}}{V_{ref}} = \frac{t_2}{t_1}$$

(1b)

1999 counts even if it represents 2.463 V, if for example that were V_{ref}.)

Making a DVM Chip Computer Compatible

There are more bus configurations than I know what to do with lately, so I set up this interface to run from decoded input and output ports. Whether they be memory mapped IO or not, we do not care, as long as the outputs are latched and the inputs can be driven by low power Schottky TTL devices.

To fully utilize this eight channel 3½ digit DVM, we must design the correct hardware interface and write a universal software driver.

Hardware and Data Format

Figure 2 details the schematic of the 8 channel interface board. IC1 is the MC14433 DVM chip. With the values chosen, it will perform approximately 25 conversions a second. Reducing the 68 K resistor between pins 10 and 11 to about 27 K will increase this to about 50 conversions per second. This is an out of specification condition and, though probably successful, is dependent on individual parts.

Each output pin of IC1 has the power to drive one LS TTL load. Since all input ports are not necessarily low power, we provide IC3 and IC4 as buffers. They are 74LS04s and while they are capable of driving regular TTL, they do invert the output data of the DVM. Any driver program must complement the BCD and digit data it receives from this interface before using it.

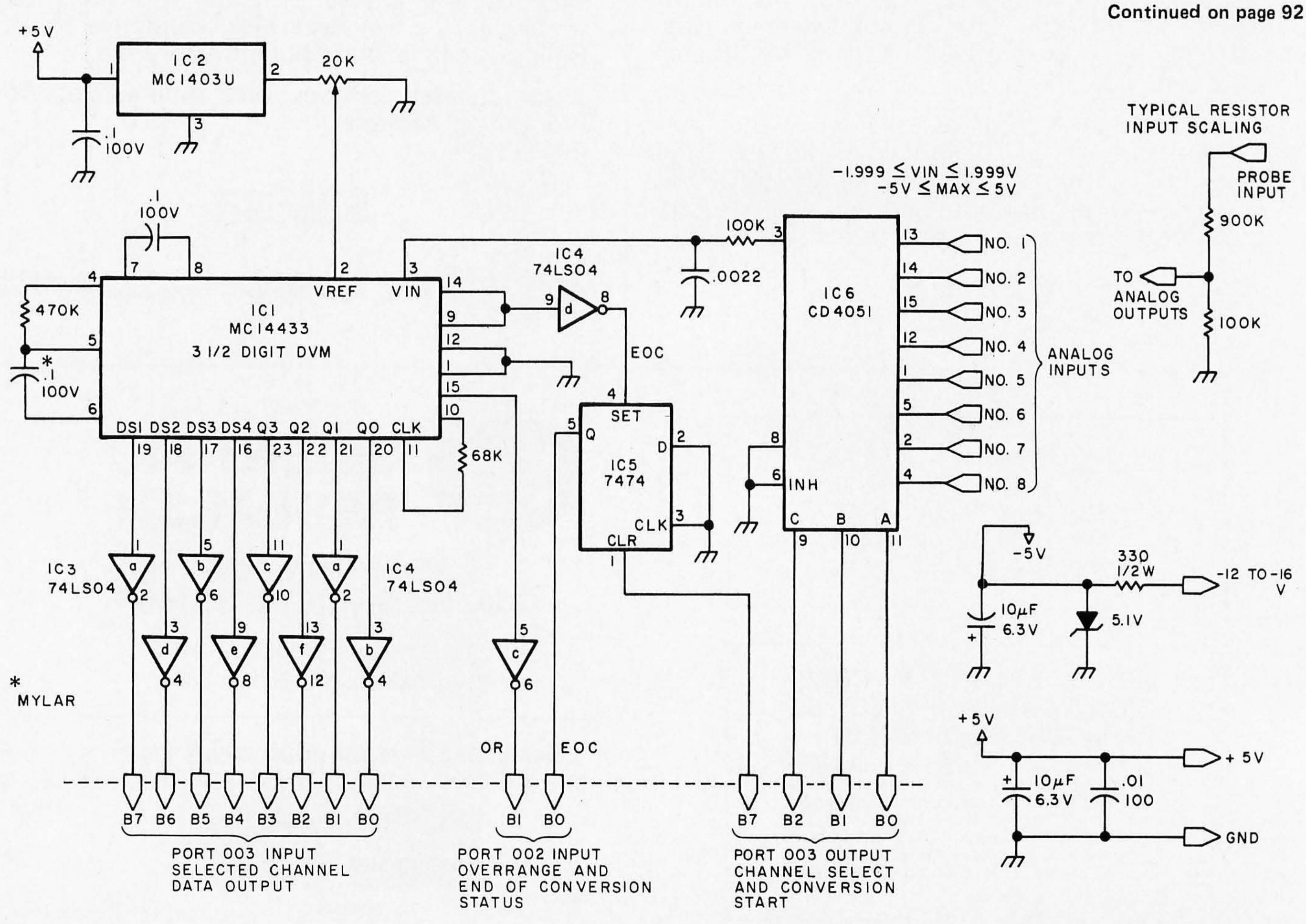
IC2 is a MC1403 precision voltage reference chip and supplies the V_{ref} input. This IC will vary only 7 mV over a range of 0° to 70°C from its nominal 2.5 V output. While a zener diode might also supply an adequate reference voltage, the temperature drift characteristics of the average zener would negate the value of a 3½ digit converter if used beyond a 5 or 10°C temperature variation. A precision voltage integrated circuit is an absolute must if this circuit is to be used for practical applications.

IC5 is a 7474 which is used here as a set-reset flip flop. The end conversion signal from IC1 sets it, and an output bit from the computer resets it after reading the output data.

IC6 is an 8 input CMOS multiplexer. Its address lines are tied directly to a latched

Continued on page 92

Figure 2: Circuit for the 8 channel 3½ digit voltmeter.



The LATEST in Tape Systems



MODEL CC-8 $175.00

4800 BAUD CASSETTE RECORDER

An ASYNCHRONOUS NRZ type Recorder with remote motor start/stop. Error rate 10^8 at 4800 BAUD. Can be used from 110 to 4800 BAUD into a UART or "Bit Banger PIA" —no clocking required. This is not an audio recorder. It takes RS232 or TTL signals from the terminal or computer and gives back the same signals. No audio interface is used. Motor start/stop is manual or through TTL or RS232 signals.

Tape speed is 3.2"/second nominal; 1.6"/sec. optional. 110 volt, 60 Hz, 5 watts. (220 Volts on special order). Can use high quality audio cassettes (Philips Type) or certified data cassettes. Can be used in remote locations from a 12 Volt battery.

Recommended for DATA LOGGING, WORD PROCESSING, COMPUTER PROGRAM RELOADING and DATA STORAGE. Especially recommended for 6800 systems, 6502 systems, 1800 systems and beginners with the 8080 systems. Manual control except for motor start/stop. 6800 or 8080 software for file or record searching available on request with order. Used by major computer manufacturers, Bell Telephone and U.S. Government for program reloading and field servicing.

AVAILABILITY – Off the shelf.

6800 CONTROLLER for SWTP

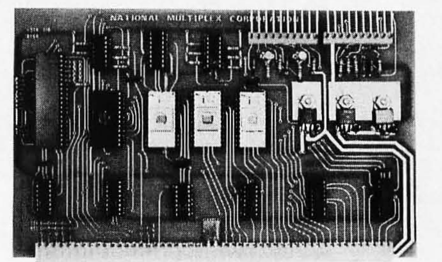
PROVIDES MONITOR AND TAPE SOFTWARE in EPROM. EXPANDS MIKBUG with 1 K of ADDITIONAL ROM PROGRAM.

This is a complete tape controller for the SWTP 6800 system. Has 3 K of EPROM space for your own programs. A 1 K ROM (2708) is provided with all tape and monitor functions. The ROM program is identical to our extensive 8080 ROM program.

Has one ACIA for one or two tape drives, one USART for an additional Serial port and a 4 bit parallel port for motor control. Will control one or two CC-8 or 3M3A drives with the software provided. Can be used with other tape drives controllable with 4 TTL bits if appropriate software changes are made.

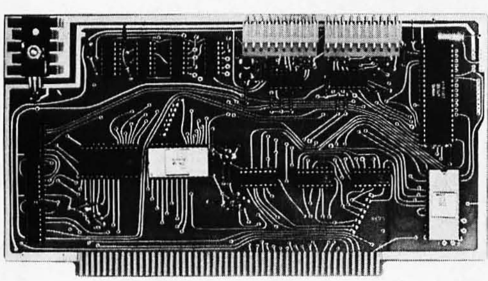
Extra serial port is provided for your use with a second terminal or printer. (RS232, TTL or 20 ma)

The ROM program supplements the MIKBUG program and is entered automatically on reset.



AVAILABILITY—Off the Shelf.

$190.00, Tested & Assmb. ($160.00, Kit)



PROVIDES MONITOR AND TAPE SOFTWARE in ROM TERMINAL and TAPE PORTS on SAME BOARD CONTROLS ONE or TWO TAPE UNITS (CC-8 or 3M3A)

This is a complete 8080, 8085, or Z80 system controller. It provides the terminal I/O (RS232, 20 mA, or TTL) and the data cartridge I/O, plus the motor controlling parallel I/O latches. Two kilobytes of on board ROM provide turn on and go control of your Altair or Imsai. NO MORE BOOTSTRAPPING. Loads and Dumps memory in hex on the terminal, formats tape cartridge files, has word processing and paper tape routines. Best of all, it has the search routines to locate files and records by means of six, five, and four letter strings. Just type in the file name and the recorder and software do the rest. Can be used in the BiSync (IBM), BiPhase (Phase encoded) or NRZ modes with suitable recorders and interfaces.

This is Revision 7 of this controller. This version features 2708 type EPROM's so that you can write your own software or relocate it as desired. One 2708 preprogrammed is supplied with the board. A socket is available for the second ROM allowing up to a full 2K of monitor programs.

Fits all S100 bus computers using 8080 or Z80 MPU's. Requires 2 MHz clock from bus. Cannot be used with audio cassettes without an interface. Cassette or cartridge inputs are RS232 level.

2SIO (R) CONTROLLER
$190.00 ($160.00 Kit)

AVAILABILITY – Off the shelf.

Z 80 BOARD for SWTP COMPUTER: Now you can use the 8080/Z80 software programs in your SWTP 6800 machine. Replaces your MPU board with a Z80 and ROM so that you are up and running with your present SWTP memory and MPC card. $200 assembled and tested. ($160 kit)

AVAILABLE—November '77.

OVERSEAS: Export Version 220 volt 50 hz. Write factory or: Megatron-Datameg, 8011 Putzbrunn, Munchen, Germany; Nippon Automation 5-16-7 Shiba, Minato-Ku, Tokyo, Japan; Hobbydata, FACK 20012, Malmo, Sweden; G. Ashbee, 172 Ifield Road, London SW 10-9ag: Trintronics, Ltd., 186 Queen Street W., Toronto, Ontario, Canada; EBASA, Enrique Barges 17, Barcelona 14, Spain; ARIES, 7, rue Saint Phillipe du Roule, 75008 Paris; Microlem 20131, Milano, Italy; Eagle Electric, Capetown, S. Africa.

For U.P.S. delivery, add $3.00 Overseas and air shipments charges collect. N.J. Residents add 5% Sales Tax. WRITE or CALL for further information. Phone Orders on Master Charge and BankAmericard accepted.

National Multiplex Corporation

3474 Rand Avenue, South Plainfield NJ 07080 Box 288 Phone (201) 561-3600 TWX 710-997-9530

Circle 82 on inquiry card.

SYNCHRO-SOUND ENTERPRISES, INC.

One-stop shopping for Hardware and Software

Everything you need in small computer systems with special emphasis on TERMINALS! Look at these units... compare price, quality, delivery, service... and you'll see why you don't have to look anyplace else!



LEAR SIEGLER ADM 3A TERMINAL

- Full addressable cursor
- Display format—24 lines of 80 characters per line
- Communications rates—75 to 19,200 Baud
- Computer interfaces —EIA standard

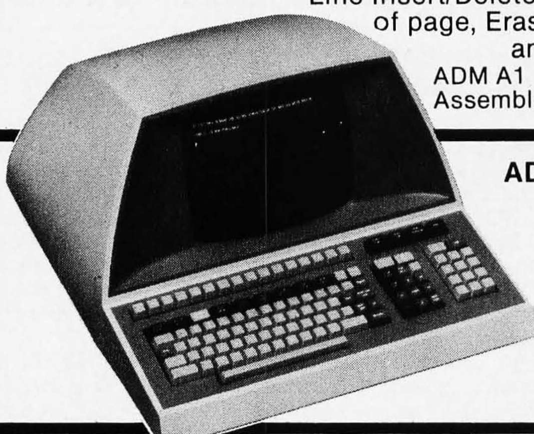
ADM 3A Kit **$739.95**
ADM 3A Assembled .. **849.95**
Lower Case option **69.00**



ADM 1A TERMINAL

- Display Format —80 characters per line by 24 lines
- Full cursor control
- Edit operations: clear screen, clear unprotected character type-over. (Options are Character Insert/Delete, Line Insert/Delete, Erase to end of page, Erase Line/Field) and much more

ADM A1 Assembled **$1398.00**



ADM 2A TERMINAL

- 1920 character display (24 x 80)
- 16 function keys for 32 commands
- Separate keyboard—119 keys
- 10 key numeric pad
- Single key edit operations
- Page, field or line edit
- Security, protected fields and much more

ADM 2A Assembled ... **$1895.00**

OKIDATA MODEL 110 LINE PRINTER

- 110 CPS dot matrix

Friction Feed... **$1149.00**
Tractor Feed..... **1279.00**
RS 232C Serial Interface ... **260.00**



OKIDATA MODEL 22 LINE PRINTER



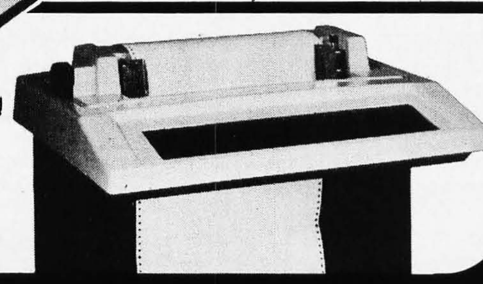
- 125 lines per minute 132-column print line
- Upper/lower case
- 8 different character sizes
- 12 IPS paper slew

Tractor Feed **$2249.00**
RS 232C Serial Interface... **379.00**

DECWRITER II

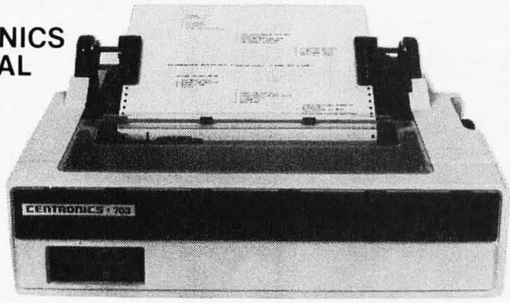
- 132 column printing
- 10-30 CPS
- Full keyboard
- Tractor feed

$1695.00



SYNCHRO-SOUND
ENTERPRISES, INC.

CENTRONICS 703 SERIAL PRINTER



- Low cost of ownership
- Bidirectional logic seeking printing
- Microprocessor electronics
- Excellent print quality

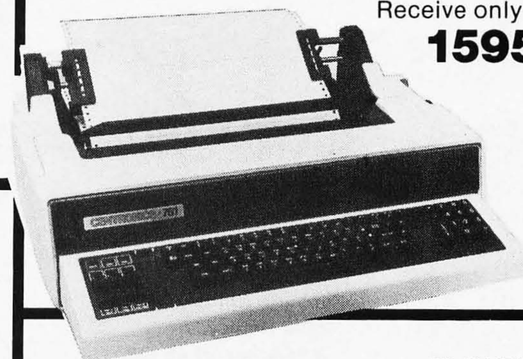
$2395.00

CENTRONICS 761 PRINTER

- 300 Baud serial transmission
- Bidirectional and incremental printing
- RS232, CCITT-V24, or current loop interface
- Baud selection (110/150/300)

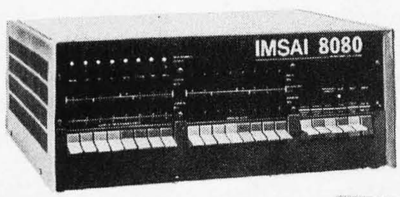
KSR with Keyboard **$1695.00**

Receive only version **1595.00**



IMSAI 8080 MICROCOMPUTER

● Powerful ● Low cost ● Easy to use



With 22 Slot Mother Board **$619.95**

With Z-80 CPU **849.95**

FLOPPY DISK SYSTEM

- Fully IBM 3740 media and format compatible
- Full formatter and controller built-in

ICOM Model FD 3712 Dual Drive System **$2795.00**



HAZELTINE 1500 VIDEO TERMINAL

- Reverse video
- 24 x 80 display
- Programmable brightness levels
- RS232 and current loop and much more



Assembled .... **$1149.00** Kit also available

HAZELTINE MODULAR 1 INTELLIGENT TERMINAL

- 1920 character display
- 8 different video levels
- Full editing capability
- Removable keyboard and much more

Assembled **$1659.00**



SPECIAL BUYS

Sorac IQ 120 Video Terminal Kit	$959.00
DEC LA 180 Printer	2769.00
Compucolor 8001 Color Computer	2595.00
ICOM Microfloppy System	989.00
North Star Microfloppy Disk Kit	599.00
2708 Eproms	27.95
Javelin 9" Video Monitor	159.95
Livermore Modem Model 76	299.00
Sol 20 with Solos Kit (limited supply)	999.00
Micropolis Model 1053MOD 2	1799.00
IMSAI AP44-44 Col. Printer Kit	329.00
TDL Xitan Alpha 1 Computer Kit	699.00

We carry a full line of the following: TDL, Centronics, Seals, Hazeltine, Micropolis, Hayden, IMSAI, Cromemco, Compucolor, Icom, Lear Siegler, Okidata, DEC, Javelin, North Star, Peripheral Vision. Same day delivery and shipping on most items. Full modern repair facilities on premises for complete servicing of everything we sell.

SYNCHRO-SOUND ENTERPRISES, INC.
The Computer People

193-25 Jamaica Avenue,
Jamaica, New York 11423
212/468-7067 TWX: 710-582-5886
Hours 9-4 daily and Saturday
Visit our new showroom
Working units on display
Dept. BB BankAmericard • Master Charge

Circle 112 on inquiry card.

Get Your System Together

John G Whitney
2405 Haisley Dr
Ann Arbor MI 48103

So you now have your own home computer system up and running. Your processor is on one bench, a cassette mass storage memory unit on another. The terminal stands nearby, with the paper tape punch and your TV video monitor on the floor. To top it off you have 20 interconnecting cables lying around. Does this remind you of your computer system? Things don't have to be this way when with a moderate amount of time and money one can convert one's computing apparatus into a well organized and laid out system.

Most professionally installed minicomputer systems have most of their hardware mounted in one rather large cabinet. Mounting the hardware in such a cabinet eliminates many interconnecting cables that otherwise would be lying around on the floor. A cabinet for your system will increase the system's reliability by shortening cable lengths between system components, since they are now all mounted together. The shorter cables will have less capacitance which will in turn lower the amount of byte transfer errors due to noise pickup. A cabinet will also provide easy movability of the total computing system without much trouble. By mounting your computer hardware in such a cabinet you can achieve these results in addition to improving your system's appearance and efficiency.

At almost any used computer surplus center one can find a fairly cheap and adequate stripped down computer frame or cabinet that will serve the purpose. I have found a 20 by 25 by 60 inch (40.8 by 63.5 by 15.24 cm) size frame quite suitable for my homebrew computing system. The size depends mainly on the size of the existing equipment in your system. Some of the better cabinets are the ones in which all four sides have panels which swing open to expose the computer hardware for easy servicing. If you fail to locate such a cabinet, you can easily construct an adequate wooden enclosure.

After acquiring a cabinet it's usually best to install a fairly large ventilating fan to keep the heat generated by the hardware to a minimum. Another addition such as a row of 110 V AC outlets, mounted on the inside of one side panel, will decrease your work when changing or increasing your computer's equipment. A main power switch for all electrical equipment in the cabinet, when mounted near the front, becomes quite handy when all power is to be disconnected. Other additions, like a smoked glass or plastic front door panel to improve your cabinet's appearance, are left to your imagination.

Not only will the cabinet provide an enclosure for your equipment, but any sufficiently large unused space can be used to store your computer software, in paper tape or cassette form. When utilizing space for your system's software it becomes necessary to insulate and shield these areas from all electrical wires or possible strong magnetic fields. This can be done either by moving the wires away from the space or by surrounding them with a steel shield which will confine the magnetic fields. Otherwise you might find that your software cassettes contain small bits of garbage.

In utilizing a cabinet with your system you won't just improve your computer's appearance and efficiency, but when someone views your system he or she won't exclaim in disbelief, "Is that the computer? That little box!"■

Tandy TC COMPUTERS

A DIVISION OF TANDY CORPORATION

ANNOUNCING
GRAND OPENING

MOST COMPLETE LINE OF MICROCOMPUTERS AND ACCESSORIES

PLACE YOUR ORDER TODAY — CALL TOLL-FREE—800-433-1679



RADIO SHACK

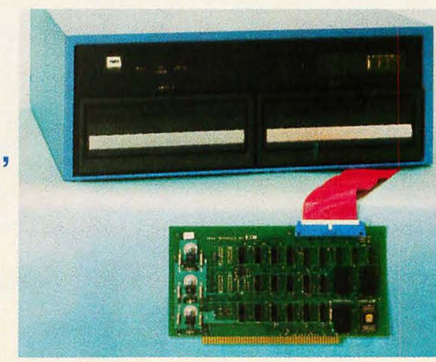


SOUTHWEST TECHNICAL PRODUCTS



CPU's — Alpha Micro, ICOM, IMSAI, Polymorphics, Processor Technology, T.D.L., Vector Graphics, Equinox

"POPULAR BRANDS" CARRIED



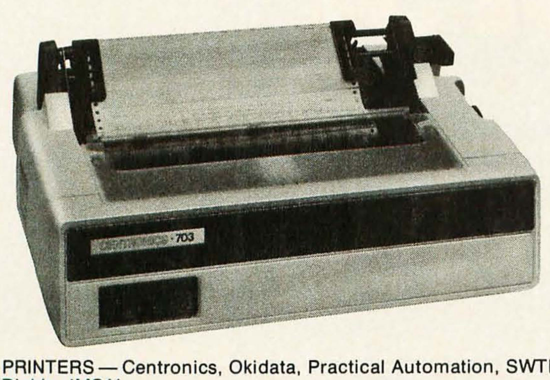
DISCS — ICOM, Digital Systems, IMSAI, Microplis, North Star, Polymorphics, Processor Technology, Shugart, Smoke Signal, SWTP, Persci, Extensys



VIDEO BRAIN



TERMINALS — Beehive, Lear Siegler, SOROC, SWTP, Informer



PRINTERS — Centronics, Okidata, Practical Automation, SWTP, Diablo, IMSAI

Complete Line of Parts and Accessories! Call for Special Prices! 800-433-1679

Mail To: TANDY COMPUTERS B12
P.O. Box 2936, Fort Worth, Texas 76102

Please Send Me TANDY COMPUTERS Full-Line 1978 Catalog

PLEASE PRINT

Name____________________ Apt. ______

Street __________________________

City_______________ State ______ Zip ______

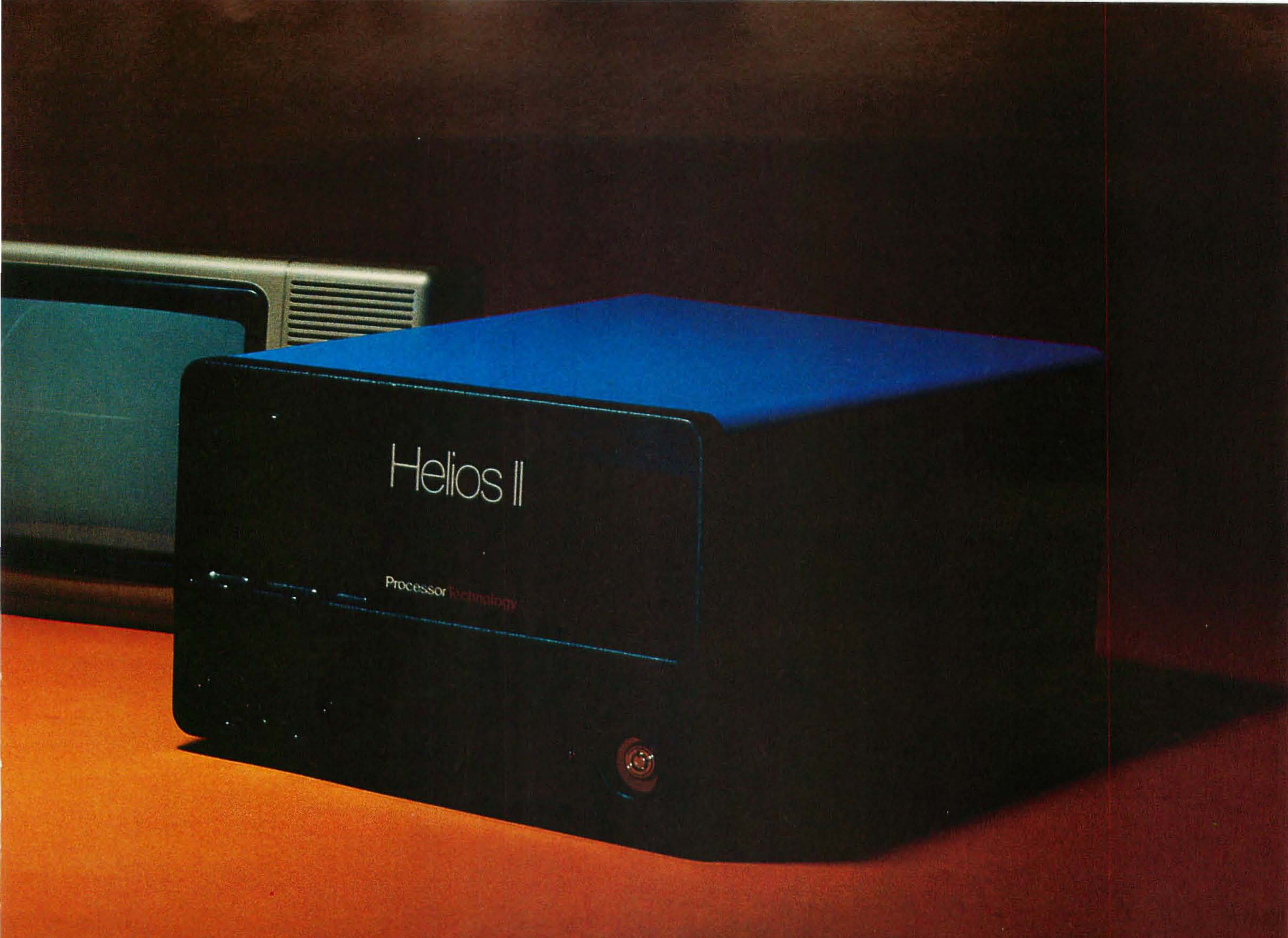


One Sol-20 equals three computers.

To do real work with any computer, big or small, it takes a complete system. That's one of the nice things about the Sol-20. It was built from the ground-up as the heart of three fixed price computer systems with all the peripheral gear and software included to get you up and on the air.

Sol System I costs just $1649 in kit form or $2129 fully burned in and tested. Here's what you get: a Sol-20 with the SOLOS personality module for stand alone computer power, an 8192 word memory, a 12″ TV/video monitor, a cassette recorder with BASIC software tape and all necessary cables.

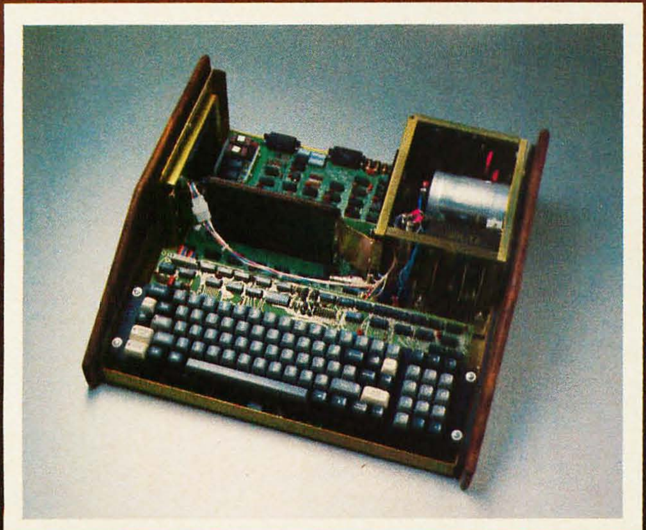
Sol System II has the same equipment plus a larger



capacity 16,384 word memory. It sells for $1883 in kit form; $2283 fully assembled.

For even more demanding tasks, Sol System III features Sol-20/SOLOS, a 32,768 word memory, the video monitor, Helios II Disk Memory System and DISK BASIC Diskette. Price, $4750 in kit form, $5450 fully assembled and tested.

And remember, though we call these small or personal computer systems, they have more power per dollar than anything ever offered. They provide performance comparable with mini-computer systems priced thousands of dollars more.

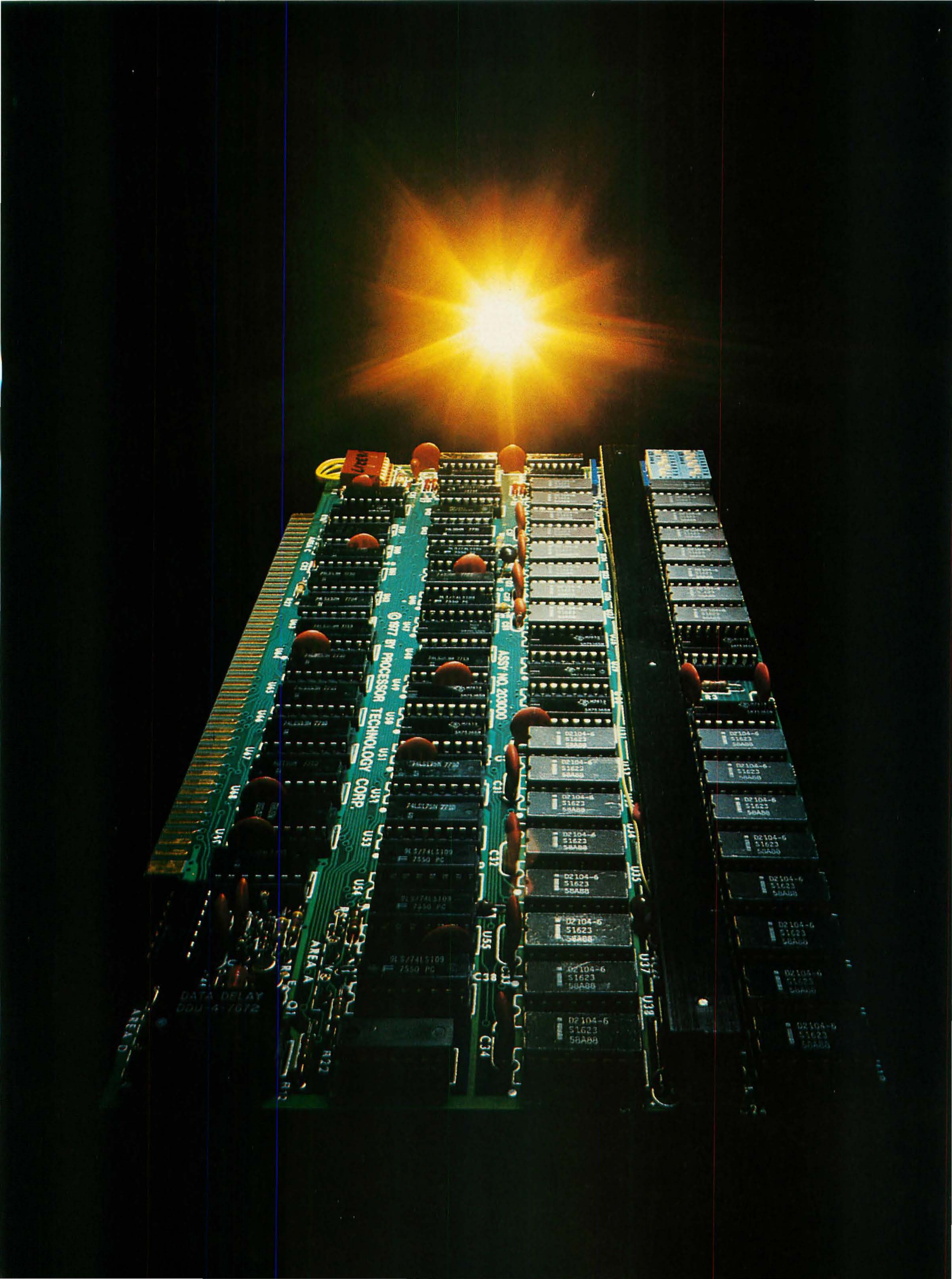


The functional beauty of Sol Computer Systems is more than skin deep. A look inside reveals a simple elegance of design and sturdy construction.

The Small Computer Catalog for the rest of the real computer system story.

Visit your local computer store for a copy of our fully illustrated 22 page catalog. Or you may write or call us if more convenient. Please address Processor Technology, Box B, 6200 Hollis Street, Emeryville, CA 94608. (415) 652-8080.

Processor Technology Corporation


© 1977 BY PROCESSOR TECHNOLOGY CORP.
ASSY NO. 203000
AREA A
DATA DELAY

The end of Kit-Kits.

The end of bad solder joints, heat damaged components and sick IC's. Introducing the Semikit. Item 1, a 16KRA Memory Board, $369.

Let's face it. Loading and soldering PC Boards is not much fun for the kit builder. Even more important, it's the place where most of the trouble gets introduced. The real fun and education comes in running and testing boards.

Now the Semikit with fully tested IC's.

At the price of a kit, Processor Technology Corporation introduces the Semikit. It's a fully stuffed, assembled and wave soldered PC Board loaded with IC's that have gone through Q.C. and final checkout (a first in the industry).

We leave you the fun of testing with our fully documented set of instructions. We do the production tasks of loading, wave soldering and inspecting the boards. You do the more interesting and time consuming chore of testing and burning-in the boards.

The result is one sweet deal for both of us. You get a board where the primary causes of damage (poor solder joints, excess solder and bad IC's) are virtually eliminated. You get a board of highest professional quality. And we get the business!

The 16KRA Memory Board's at your dealer now.

Your Processor Technology dealer has the first Semikit, a 16KRA Memory Board, in stock and ready to go right now. You can take it home tonight for $369 as a Semikit or for $399 fully assembled, tested and burned-in.

You'll have a 16,384 byte memory with a better price performance ratio than anything on the market today. Now you can afford to add quality, high density memory to your system for remarkably little. And you can add enough to solve complex computing problems right in the main frame.

The memory features invisible refresh. There's no waiting while the CPU is running. Worst case access time is 400 nsec. Each 4,096 word block is independently addressable for maximum system flexibility. Power is typically 5 watts, the same as most single 4K memory modules. Back-up power connection is built-in.

Other Semi's are coming your way.

The 16KRA Memory is Processor's first step in adding more fun, capability and reliability to your computer system at lower cost. Other modules are on the way to your dealer now. Come on down today.

Or you may contact us directly. Please address Processor Technology Corporation, Box B, 7100 Johnson Industrial Drive, Pleasanton, California 94566. Phone (415) 829-2600.

Processor Technology

Circle 95 on inquiry card.

See Sol here...

ARIZONA

Byte Shop Tempe
813 N. Scottsdale Rd.
Tempe, AZ 85281
(602) 894-1129

Byte Shop Phoenix
12654 N. 28th Dr.
Phoenix, AZ 85029
(602) 942-7300

Byte Shop Tucson
2612 E. Broadway
Tucson, AZ 85716
(602) 327-4579

CALIFORNIA

The Byte Shop
1514 University Ave.
Berkeley, CA 94703
(415) 845-6366

Computer Center
1913 Harbor Blvd.
Costa Mesa, CA 92627
(714) 646-0221

DCI Computer Systems
4670 N. El Capitan
Fresno, CA 93711
(209) 266-9566

The Byte Shop
1122 "B" Street
Hayward, CA 94541
(415) 537-2983

The Byte Shop
16508 Hawthorne Blvd.
Lawndale, CA 90260
(213) 371-2421

The Computer Mart
633-B West Katella
Orange, CA 92667
(714) 633-1222

Byte Shop
496 South Lake Ave.
Pasadena, CA 91101
(213) 684-3311

Micro-Computer
Application Systems
2322 Capitol Avenue
Sacramento, CA 95816
(916) 443-4944

The Computer Store
of San Francisco
1093 Mission Street
San Francisco, CA 94103
(415) 431-0640

Byte Shop
321 Pacific Ave.
San Francisco, CA 94111
(415) 421-8686

The Byte Shop
2626 Union Avenue
San Jose, CA 95124
(408) 377-4685

The Computer Room
124H Blossom Hill Rd.
San Jose, CA 95123
(408) 226-8383

The Byte Shop
509 Francisco Blvd.
San Rafael, CA 94901
(415) 457-9311

The Byte Shop
3400 El Camino Real
Santa Clara, CA 95051
(408) 249-4221

Recreational Computer
Centers
1324 South Mary Ave.
Sunnyvale, CA 94087
(408) 735-7480

Byte Shop of Tarzana
18423 Ventura Blvd.
Tarzana, CA 91356
(213) 343-3919

Computer Components
5848 Sepulveda Blvd.
Van Nuys, CA 91411
(213) 786-7411

The Byte Shop
2989 North Main St.
Walnut Creek, CA 94596
(415) 933-6252

Byte Shop
14300 Beach Blvd.
Westminster, CA 92683
(714) 894-9131

COLORADO

Byte Shop
3101 Walnut St.
Boulder, CO 80301
(303) 449-6233

Byte Shop
3464 S. Acoma St.
Englewood, CO 80110
(303) 761-6232

FLORIDA

Byte Shop of
Fort Lauderdale
1044 East Oakland Park
Blvd.
Ft. Lauderdale, FL 33334
(305) 561-2983

Byte Shop of Miami
7825 Bird Road
Miami, FL 33155
(305) 264-2983

Microcomputer
Systems Inc.
144 So. Dale Mabry Hwy.
Tampa, FL 33609
(813) 879-4301

GEORGIA

Atlanta Computer Mart
5091-B Buford Hwy.
Atlanta, GA 30340
(404) 455-0647

ILLINOIS

Champaign Computer
Company
318 N. Neil Street
Champaign, IL 61820
(217) 359-5883

itty bitty machine co.
1322 Chicago Ave.
Evanston, IL 60201
(312) 328-6800

itty bitty machine co.
42 West Roosevelt
Lombard, IL 60148
(312) 620-5808

INDIANA

The Data Domain
406 So. College Ave.
Bloomington, IN 47401
(812) 334-3607

The Byte Shop
5947 East 82nd St.
Indianapolis, IN 46250
(317) 842-2983

The Data Domain
7027 N. Michigan Rd.
Indianapolis, IN 46268
(317) 251-3139

IOWA

The Computer Store
of Davenport
4128 Brady Street
Davenport, IA 52806
(319) 386-3330

KENTUCKY

The Data Domain
3028 Hunsinger Lane
Louisville, KY 40220
(502) 456-5242

MICHIGAN

The Computer Store
of Ann Arbor
310 East Washington
Ann Arbor, MI 48104
(313) 995-7616

Computer Mart
of Royal Oak
1800 W. 14 Mile Rd.
Royal Oak, MI 48073
(313) 576-0900

General Computer Store
2011 Livernois
Troy, MI 48084
(313) 362-0022

MINNESOTA

Computer Depot, Inc.
3515 W. 70th St.
Minneapolis, MN 55435
(612) 927-5601

NEW JERSEY

Hoboken Computer Works
No. 20 Hudson Place
Hoboken, NJ 07030
(201) 420-1644

The Computer Mart
of New Jersey
501 Route 27
Iselin, NJ 08830
(201) 283-0600

NEW YORK

The Computer Shoppe
444 Middle Country Rd.
Middle Island, NY 11953
(516) 732-4446

The Computer Mart
of New York
118 Madison Ave.
New York, NY 10001
(212) 686-7923

The Computer Corner
200 Hamilton Ave.
White Plains, NY 10601
(914) 949-3282

NORTH CAROLINA

ROMs 'N' RAMs
Crabtree Valley Mall
Raleigh, NC 27604
(919) 781-0003

OHIO

Byte Shop
2432 Chester Lane
Columbus, OH 43321
(614) 486-7761

Computer Mart of Dayton
2665 S. Dixie Ave.
Dayton, OH 45409
(513) 296-1248

OREGON

Byte Shop Computer Store
3482 SW Cedar Hills Blvd.
Beaverton, OR 97005
(503) 644-2686

The Real Oregon
Computer Co.
205 West 10th Ave.
Eugene, OR 97401
(503) 484-1040

Byte Shop Computer Store
2033 SW 4th Ave.
Portland, OR 97201
(503) 223-3496

RHODE ISLAND

Computer Power, Inc.
M24 Airport Mall
1800 Post Rd.
Warwick, RI 02886
(401) 738-4477

SOUTH CAROLINA

Byte Shop
2018 Green St.
Columbia, SC 29205
(803) 771-7824

TENNESSEE

Microproducts & Systems
2307 E. Center Street
Kingsport, TN 37664
(615) 245-8081

TEXAS

Computer Port
926 N. Collins
Arlington, TX 76011
(817) 469-1502

Computertex
2300 Richmond Ave.
Houston, TX 77006
(713) 526-3456

Interactive Computers
7646½ Dashwood Rd.
Houston, TX 77036
(713) 772-5257

Neighborhood Computer
Store
#20 Terrace
Shopping Center
4902 - 34th Street
Lubbock, TX 79410
(806) 797-1468

The Micro Store
634 So. Central
Expressway
Richardson, TX 75080
(214) 231-1096

VIRGINIA

The Computer Systems
Store
1984 Chain Bridge Rd.
McLean, VA 22101
(703) 821-8333

Media Reactions Inc.
Reston International Center
11800 Sunrise Valley Dr.
Suite #312
Reston, VA 22091
(703) 471-9330

The Home Computer Center
2927 Virginia Beach Blvd.
Virginia Beach, VA 23452
(804) 340-1977

WASHINGTON

Byte Shop Computer Store
14701 N.E. 20th Ave.
Bellevue, WA 98007
(206) 746-0651

The Retail Computer Store
410 N.E. 72nd
Seattle, WA 98115
(206) 524-4101

WISCONSIN

The Milwaukee
Computer Store
6916 W. North Ave.
Milwaukee, WI 53213
(414) 259-9140

WASHINGTON D.C.

Georgetown
Computer Store
3286 M Street NW
Washington, D.C. 20004
(203) 362-2127

CANADA

Trintronics
160 Elgin St.
Place Bell Canada
Ottawa, Ontario K2P 2C4
(613) 236-7767

Computer Mart Ltd.
1543 Bayview Ave.
Toronto, Ontario M1K 4K4
(416) 484-9708

First Canadian
Computer Store Ltd.
44 Eglinton Ave. West
Toronto, Ontario M4R 1A1
(416) 482-8080

The Computer Place
186 Queen St. West
Toronto, Ontario M5V 1Z1
(416) 598-0262

Basic Computer Group Ltd.
1548 East 8th Ave.
Vancouver, B.C. V6J 4R8
(604) 736-7474

Pacific Computer Store
4509 Rupert St.
Vancouver, B.C. V5R 2J4
(604) 438-3282

ProcessorTechnology

Circle 95 on inquiry card.

I may have bugs in the program, but the programmer's got bats in his belfry.

Jack and the Machine Debug

. . . or Reading the Traces of the Wild Program

Robert D Grappel
148 Wood St
Lexington MA 02173

Jack E Hemenway
151 Tremont St
Boston MA 02116

"It has to be done by now. That subroutine can't take much more than a few milliseconds per entry, and there aren't many entries. I'll give it a few more seconds." Jack sat nervously puffing his cigar. "It can't take this long," said Jack, his patience exhausted. He punched the RESET button.

"What do you want now, Jack? Here I am, faithfully running your program, and you interrupt me. Find a mistake in your code?"

"Hardly. You should be done by now. What have you been doing that took so long?"

"Well, when you interrupted me, I think I was executing a load-immediate instruction."

"Where?"

"How should I know? You interrupted me. I'm in the monitor ROM now. I can't keep track of every instruction I execute."

"True, true. It sure would be nice if you could, though."

"Well, I can't. I already assemble your programs for you; you can't expect me to debug them for you too! That's supposed to be your department!"

"I know, computer. How do I figure out where you went wrong?"

"How do I know?"

"Calm yourself or I'll use your parts in my F8."

"Okay, Jack. I'm sorry I lost my head. Anything would be better than inflicting that F8 on us. How about trying a breakpoint?"

"Good idea! Computer, sometimes you amaze me. Try a breakpoint at the subroutine return."

"Shouldn't I reload the program first, Jack?"

"I guess so." Jack waited as computer reloaded the program from its cassettes. "Now, put a software interrupt at 1FC0."

"One SWI inserted (hexadecimal 3F to me). Shall I run the program now?"

"Start." Jack went into the kitchen for a beer. He returned a few minutes later. "Computer! What are you doing? RESET!"

"Now what?"

"I told you to set a breakpoint!"

"I did set a breakpoint; see the 3F at 1FC0. I just haven't executed that instruction yet."

"Why not?"

"I haven't the foggiest idea. I just execute them in the order that you wrote them. Writing programs is supposed to be your contribution to our work."

"Don't get snide. Remove the breakpoint."

"Done."

"Now, put the breakpoint at 1FA2."

"I'll reload the program first, Jack."

"I guess you should, but I hate waiting for those cassettes."

"They're your design, remember. If you want speed, buy me some disks."

"They're on order."

"Great. Now let me load the program the best I can from these archaic, cranky, slow, old . . ."

"Just do the job without the commentary!"

The cassette in the read drive turned ever so slowly. "I'm ready now, Jack. The breakpoint is set."

Continued on page 133

I just execute them in the order that you wrote them. Writing programs is supposed to be your contribution to our work . . .

Continued from page 80

output port. The usual conversion sequence is to set the channel information to the multiplexer, clear the EOC flip flop and wait for an end of conversion signal. More on this later.

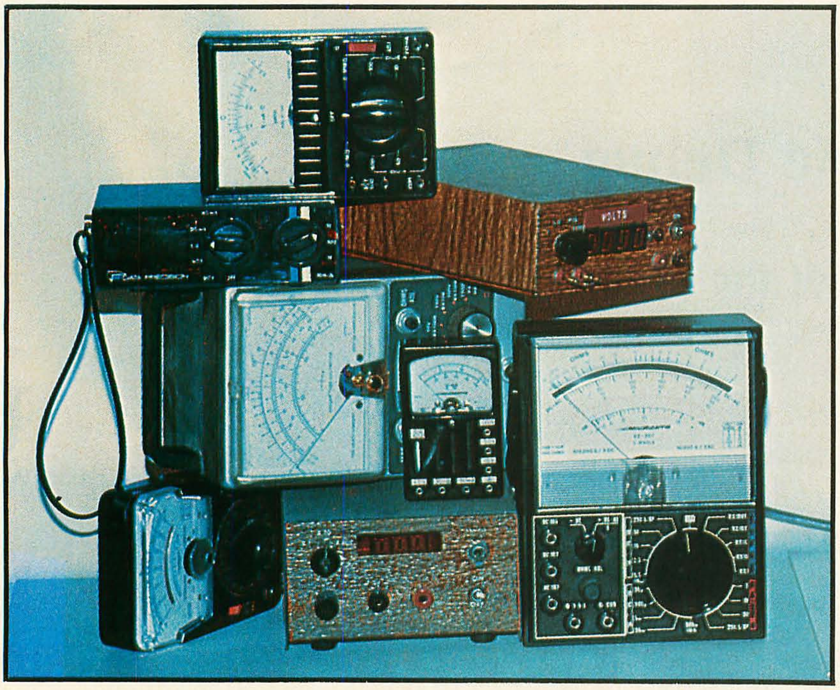


Photo 2: *Eight meters (some are multimeters, others are voltmeters) which could be replaced (at least for DC voltage measurements) by the computerized 8 channel voltmeter described here.*

Data Format

As I stated earlier, the data from the DVM to the computer is both serial and parallel. There are four digit select lines and four BCD data lines (see table 1).

With respect to what the computer sees through the 74LS04 buffers, the digit select output is low when the respective digit is selected. The most significant digit (½ digit DS1) goes low immediately after an EOC pulse, followed by the remaining digits sequencing from most significant to least significant digit (MSD to LSD). An interdigit blanking time of two clock periods is included to ensure that the BCD data has settled. The multiplex clock rate is equal to the system clock frequency divided by 80.

During the ½ digit (DS1), the polarity and certain status bits are available. It would be confusing to list the status bits, since they are not being used in this application for autoranging. The polarity will be Q_2 and a "1" will indicate negative. The ½ digit value will appear on Q_3 and a "1" will indicate high.

The interface is summarized by port

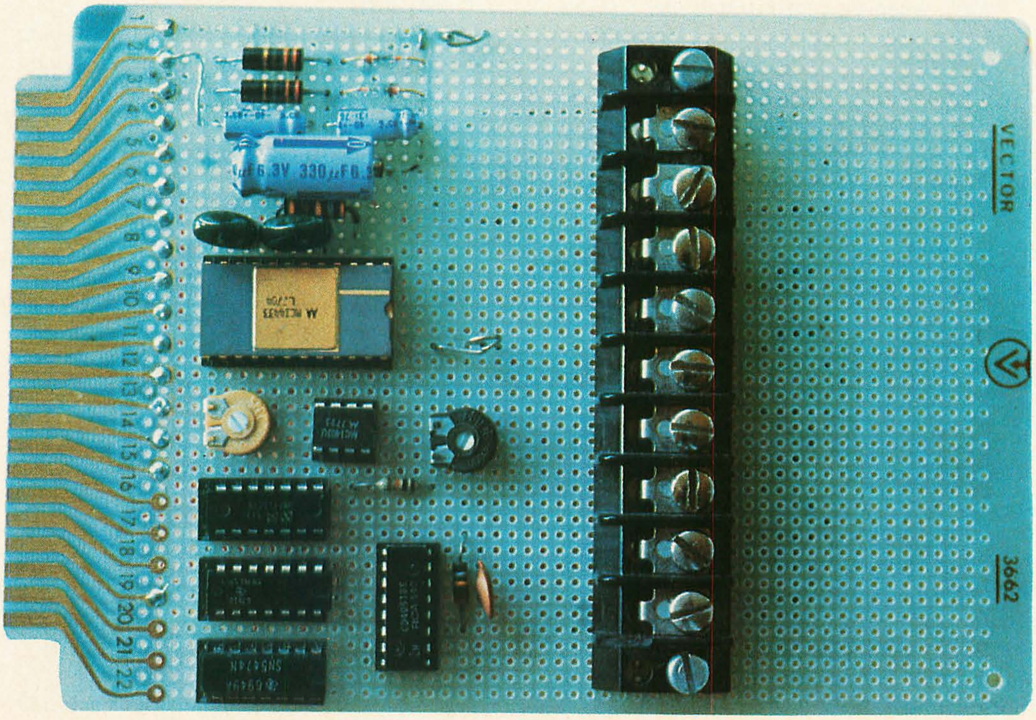
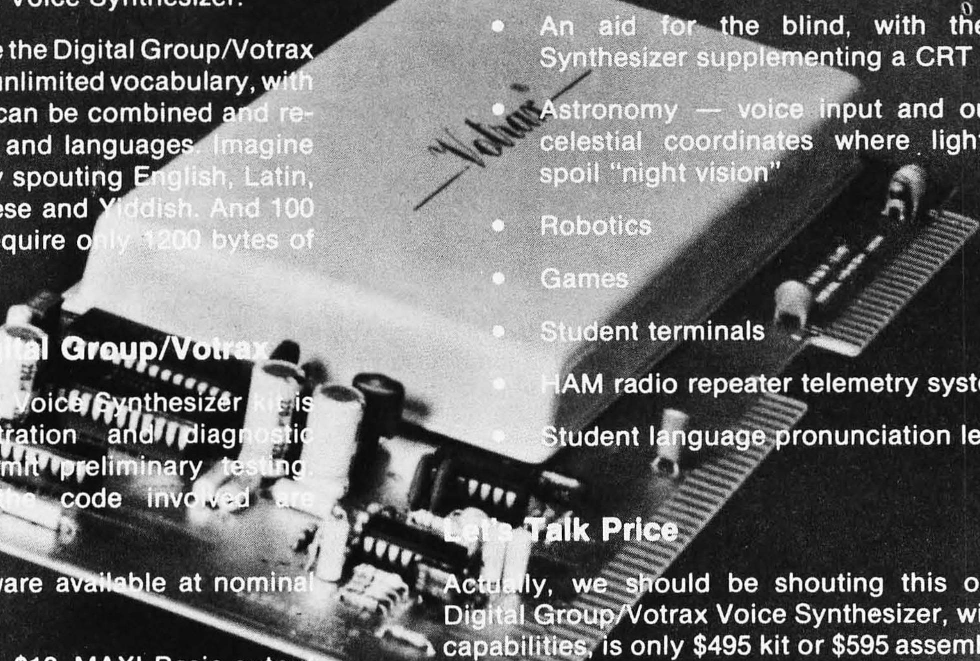


Photo 3: Prototype board for the 8 channel 3½ digit voltmeter.

Talk to our Computer... and it will talk back!

(Plainly speaking, it's only from the Digital Group.)



Now, your Digital Group computer becomes more than a silent partner. You can vocally command your computer . . . it will listen . . . and it will talk back to you. How? With the introduction of the exciting new Digital Group/Votrax Voice Synthesizer.

All this is possible because the Digital Group/Votrax Voice Synthesizer has an unlimited vocabulary, with 64 "human sounds" that can be combined and re-combined to form words and languages. Imagine your own computer glibly spouting English, Latin, Spanish, Russian, Japanese and Yiddish. And 100 average English words require only 1200 bytes of memory!

Programming the Digital Group/Votrax

The Digital Group/Votrax Voice Synthesizer kit is supplied with demonstration and diagnostic software which will permit preliminary testing. Assembler listings of the code involved are included.

We have additional software available at nominal cost:

- "Talking Basic" — $10. MAXI-Basic output converted to English.
- "Talking CW" — $10. For impressing your HAM buddies. Requires the forthcoming HAM interface card.
- "Latin and Spanish Talking" — $10. Hear the computer repeat letters and words typed in Latin or Spanish.
- Demonstration Tape — $5. A sample of audio tape and a complete explanation of the system.

Bonus: A basic input circuit is included that may be programmed to understand a small vocabulary of voice commands.

Unlimited Applications

Consider these possibilities:

- An aid for the blind, with the Voice Synthesizer supplementing a CRT display
- Astronomy — voice input and output of celestial coordinates where light would spoil "night vision"
- Robotics
- Games
- Student terminals
- HAM radio repeater telemetry systems
- Student language pronunciation learning

Let's Talk Price

Actually, we should be shouting this one. The Digital Group/Votrax Voice Synthesizer, with all its capabilities, is only $495 kit or $595 assembled and tested. That's language anybody can understand.

O.K., you've listened briefly to what we have to say about the new Digital Group/Votrax Voice Synthesizer. But we can keep right on talking! Write or call today for *all* the details — music to your ears.

the digital group Inc.

po box 6528 denver, colorado 80206 (303) 777-7133

Circle 38 on inquiry card.

allocations in table 1. (Note: I have assigned particular port numbers to each byte. These designations will run directly with the software driver provided. If the reader wishes to assign different port numbers, that is fine, but remember to modify the driver software to reflect the changes.)



Photo 4: An illustration of the accuracy of the computerized voltmeter. A Data Precision 4½ digit digital multimeter and the author's system simultaneously measure a C cell battery. The computer value is 1.540 V compared with the Data Precision reading of 1.5402 V.

Designing an Analog to Digital Converter Software Driver

For a hardware personality like me, software is a tedious task. I don't like writing any more than I have to and if it is possible to write a universal piece of code which is compatible with any operating system, all the better. Units such as the digital to analog converter I presented in the September 1977 BYTE (page 30) do not need software drivers because the hardware is explicitly designed to be independent of computer timing. *Timing* is the key word. A "software driver" is the same as its hardware counterpart. Both serve to couple the computer to external devices and synchronize the timing. The most obvious driver already existing in a computer system like my Digital Group system is the asynchronous data link to the tape cassette, video display and printer. The computer is instructed through this program to perform explicitly timed operations which result in the correct serial input and output.

The 3½ digit DVM interface is not unlike a communications driver. To effectively obtain data from the interface, the computer

Table 1: IO port data formats.

Command Output Byte (Port 003 OUT) (Enable = 1 Disable = 0)

- B7 = EOC/Interrupt Enable/disable
- B6, B5, B4, B3: Future Expansion
- B2, B1, B0: Channel Select, 0-7

Status Input Byte (Port 002 IN)

- B7, B6, B5, B4, B3, B2: Not Used
- B1 = Out of Range ($-1.999 < V_{in} > 1.999$)
- B0 = End of Conversion

Data Input Byte (Port 003 IN)

Bit	When true =	Symbol	IC1 Pin Number
B7 = 1st digit (MSD)	B7→0	DS1	19
B6 = 2nd digit	B6 (N/A)	DS2	18
B5 = 3rd digit	B5 (N/A)	DS3	17
B4 = 4th digit	B4 (N/A)	DS4	16
B3 (BCD Digit Value)	B3 = 1/2 digit value	Q_3	23
B2 (BCD Digit Value)	B2 = Polarity	Q_2	22
B1 (BCD Digit Value)	B1 = N/A	Q_1	21
B0 (BCD Digit Value)	B0 = Status Bit	Q_0	20

SCELBI's new '8080' STANDARD ASSEMBLER

FUNCTION: Assembles programs written in symbolic language for an 8080 CPU on an 8080 based system.

HARDWARE REQUIRED: 8080 computer with minimum of 4K memory (of which at least 1K should be RAM); a source listing input device; an object code output device.

OPTIONAL HARDWARE: A system console device such as a keyboard/CRT or keyboard/printer will allow convenient control of the program using executive commands; additional memory beyond 4K will allow expanded symbol table length, or capability to assemble directly into memory.

SOFTWARE REQUIRED: User provided I/O driver routines for whatever I/O devices will be utilized. Each I/O device is linked to the program by a *single* vector for ease in adapting the program to individual systems.

MEMORY UTILIZED: The assembled listing provided in the manual resides in pages 01 through 0A (hexadecimal – 001 through 012 octal). Pages 00, part of 0A, all of 0B and 0C (hexadecimal – 000, part of 012, 013 and 014 octal) are left available for user provided I/O routines. Pages 0D (hexadecimal – 015 octal) on up used for symbol table storage (or as direct assembly areas in systems with sufficient memory).

MNEMONICS UTILIZED: This program is written in, and accepts for assembly purposes, standard industry accepted mnemonics for the 8080 CPU (such as MOV A,B; INX H: CALL; etc.) [Note: SCELBI is discontinuing its use of special 8008 compatible mnemonics which have characterized its 8080 programs in the past.]

PSEUDO-OPERATORS: Accepts the ORG (originate), END (stop assembly), SET (define a name), DB (data byte), DS (data string) and DW (data word or double byte) pseudo-operators.

PROGRAM OPERATION: The program processes a source listing in two passes to produce assembled object code. An optional third pass allows an assembled listing to be obtained. Listings may be obtained in hexadecimal or octal format. The program will also display the contents of the symbol table at the operators request. The program can process source listings as single or multiple files. Program operation may be controlled from a console device using executive commands or through computer panel switches by jumping to appropriate locations within the program.

SOURCE FORMAT: Convenient, easy to use, variable length fields permitted. Labels may be 1 to 6 characters in length, accepts both hexadecimal and octal numbers with or without leading zeros, has "literal" capability (can accept ASCII characters directly as data), allows use of letters of numbers as CPU register operands.

DOCUMENTATION: Thorough – in the SCELBI tradition! The program manual describes the operation of the assembler, presents detailed discussions of all major routines, and contains two completely assembled listings (one provided in hexadecimal and one in octal notation). Of course it includes operating instructions and even provides a routine that may be used for loading programs produced by the assembler!

SPECIAL FEATURES: Because the program has been carefully organized and written with all memory references assigned labels, it may be readily reassembled to reside in any general area in memory. It may even be reassembled to reside in ROM provided that some RAM area is available for scratch pad and symbol table use!

OPTIONS: A punched paper tape of the object code for this assembler (as described in the documentation) is available. The object code tape is provided in the widely accepted "hexadecimal format." Also, the complete, commented source listing of the program as presented in the documentation is available in straight ASCII format on punched paper tape. Fan-fold paper tapes are provided for ease in handling. Additionally, opaque paper tape is supplied to facilitate the use of low cost optical paper tape readers now in widespread use. NOTE: Paper tapes are sold only as optional supplements to the documentation.



Scelbi's 8080 Standard Assembler: **$19.95** Optional object code on punched paper tape, specify 8080SA-OPT: **$10.00.** Optional commented source listing on punched paper tape, specify 8080SA-SPT: **$39.00.**

OPTIONAL PAPER TAPE NOW AVAILABLE!

SCELBI COMPUTER CONSULTING INC.
Post Office Box 133 PP STN
Milford, CT 06406 Dept. B

Circle 98 on inquiry card.

Listing 1: An assembly program for driving the 8 channel 3½ digit voltmeter in figure 3. It is designed to run on the Z-80 and is assembled to occupy memory page octal 140.

```
ASSM 140000 140000

140000                    *
140000                    *** MC14433 3 1/2 DIGIT A/D CONVERTER DRIVER
140000                    *
140000                    * REV 1.6
140000                    *
140000                    DIP    EQU  3      DATA INPUT PORT NUMBER
140000                    SIP    EQU  2      STATUS INPUT PORT NUMBER
140000                    COP    EQU  3      COMMAND OUTPUT PORT NUMBER
140000                    EEOC   EQU  200    ENABLE EOC INPUT
140000                    DEOC   EQU  000    DISABLE EOC INPUT
140000                    *
140000                    *
140000                    * CONVERTED CHANNEL DATA BUFFERS
140000                    *
140000 000 000            CHAN0  DW   000000
140002 000 000                   DW   000000
140004 000 000            CHAN1  DW   000000
140006 000 000                   DW   000000
140010 000 000            CHAN2  DW   000000
140012 000 000                   DW   000000
140014 000 000            CHAN3  DW   000000
140016 000 000                   DW   000000
140020 000 000            CHAN4  DW   000000
140022 000 000                   DW   000000
140024 000 000            CHAN5  DW   000000
140026 000 000                   DW   000000
140030 000 000            CHAN6  DW   000000
140032 000 000                   DW   000000
140034 000 000            CHAN7  DW   000000
140036 000 000                   DW   000000
140040                    *
140040                    * INTERMEDIATE DATA BUFFERS
140040                    *
140040 000                POLVAL DB   000    LAST POLARITY VALUE (0=POSITIVE)
140041 000                CHAN   DB   000    CURRENT CHANNEL NUMBER
140042 000 000            CCP    DW   000000 COMMAND CHANNEL PARAMETER
140044 000 000            STATUS DW   000000 RETURN STATUS PARAMETER
140046                    *
140046                    *
140046                    *** START A/D CONVERTER
140046                    *
140046                    *     INPUT PARAMETER=DE REGISTER WITH CHANNEL SELECT BITS
140046                    *                     SET FOR DESIRED CHANNEL (BIT 0=1
140046                    *                     FOR CHANNEL 0, ETC.)
140046                    *     OUTPUT PARAMETER=HL REGISTER(BIT 0 FOR CHANNEL 0
140046                    *                      WHERE 0=GOOD VALUE;1=OUT OF RANGE)
140046                    *
140046 353                START  EX   DE,HL  SAVE INPUT PARAMETER
140047 042 042 140               LD   (CCP),HL
140052 257                       XOR  A      INITIALIZE CHANNEL NUMBER
140053 375 041 040 140           LD   IY,POLVAL INITIALIZE INTERMEDIATE DATA POINTER
140057 375 167 001               LD   (IY+1),A ZERO CHANNEL NUMBER
140062                    *
140062                    * START A/D CONVERTER AND ESTABLISH POLARITY
140062                    *
140062 006 002                   LD   B,2    CYCLE TWO TIMES
140064 076 007                   LD   A,7    SELECT CHANNEL 8
140066 323 003            AGAIN  OUT  COP    SELECT CHANNEL
140070 366 200                   OR   EEOC
140072 323 003                   OUT  COP    ENABLE EOC INPUT
140074 333 002            WAIT   IN   SIP    READ STATUS
140076 313 107                   BIT  0,A    TEST FOR EOC
140100 050 372                   JR   Z,WAIT JUMP IF NOT TRUE
140102 020 362                   DJNZ AGAIN  JUMP IF NOT DONE
140104 006 200                   LD   B,200  SELECT DGIT 1
140106 315 361 140               CALL RDIG   READ DIGIT
140111 016 000                   LD   C,0    POLARITY=POSITIVE
140113 313 122                   BIT  2,D    TEST POLARITY BIT
140115 040 001                   JR   NZ,POS JUMP IF POSITIVE
140117 014                       INC  C      POLARITY=NEGATIVE
140120 375 161 000        POS    LD   (IY+0),C SAVE CURRENT POLARITY
140123                    *
140123                    * SELECT NEXT CHANNEL FOR CONVERSION
140123                    *
140123 072 042 140        SELNXT LD   A,(CCP) LOAD CHANNEL COMMAND PARAMETER
140126 313 077                   SRL  A      TEST NEXT CHANNEL BIT
140130 062 042 140               LD   (CCP),A RESTORE
140133 070 010                   JR   C,SEL001 JUMP IF CHANNEL SELECTED
140135 312 355 140               JP   Z,RAPUP
140140 375 064 001        INCCN  INC  (IY+1) INCREMENT CHANNEL NUMBER
140143 030 356                   JR   SELNXT
140145 335 041 000 140    SEL001 LD   IX,CHAN0 LOAD DATA BUFFER BASE ADDRESS
140151 026 000                   LD   D,0
140153 375 136 001               LD   E,(IY+1) LOAD CURRENT CHANNEL NUMBER
140156 313 043                   SLA  E      CALCULATE BUFFER OFFSET
140160 313 043                   SLA  E
140162 335 031                   ADD  IX,DE
140164                    *
140164                    * SELECT CHANNEL AND START CONVERSION
140164                    *
140164 072 041 140        SCSC   LD   A,(CHAN) LOAD CHANNEL NUMBER
140167 323 003                   OUT  COP    SELECT CHANNEL
140171 366 200                   OR   EEOC   ENABLE EOC OUTPUT
140173 323 003                   OUT  COP    COMMAND A/D CONVERTER
140175                    *
140175                    * WAIT FOR EOC
140175                    *
140175 333 002            WEOC   IN   SIP    READ CONVERTER STATUS
140177 313 107                   BIT  0,A    TEST FOR EOC
140201 050 372                   JR   Z,WEOC JUMP IF NOT READY
140203 313 117                   BIT  1,A    TEST FOR OVERANGE
140205 040 124                   JR   NZ,OVER JUMP IF TRUE
140207                    *
140207                    * CONVERSION DONE;PROCESS FIRST (MSD) DIGIT
140207                    *
140207 006 200            MSD0   LD   B,200  SELECT DIGIT 1
140211 315 361 140               CALL RDIG   WAIT AND READ DIGIT 1
140214 057                       CPL
140215 017                       RRCA RIGHT  JUSTIFY DIGIT VALUE
140216 017                       RRCA
140217 017                       RRCA
140220 346 001                   AND  1      ISOLATE
140222 036 000                   LD   E,0    INITIALIZE STATUS BYTE
```

must synchronize itself to the integrated circuit and perform a set instruction repertoire to demultiplex the input data stream. There is a certain trade-off between hardware and software. Another ten or 15 chips could be added to the interface board so that it requires no more software than the digital to analog converter board, but the cost justification is not there.

Driver programs can be triggered by either a poll from another program or an interrupt which initiates execution. While both can be equally effective in certain applications, using interrupt initiated drivers which give the appearance of simultaneous computer operation can be hazardous. By now, most experimenters have mastered BASIC and are trying to find more challenging applications. But consider for a moment the BASIC interpreters most systems are provided with. They may execute divinely, but they have no source listing and therefore cannot be modified very easily. If a program utilizes information provided through interrupt driven peripherals, but has no way of knowing when the information will arrive, it is of no use. Attempting to add interrupt analog data acquisition to unsourced sequentially interpreted BASIC is more than I intend to explain this month. Sometime in a future article I'll describe a control application which uses interrupts as they were intended.

Adding this DVM interface to BASIC requires a polled driver. A machine language program is written which can be inserted anywhere in the computer's memory (assuming it's assembled to execute there, of

Constructing the Interface

1. Use IC sockets and solder in all passive components.
2. Turn on the power and ensure that the correct supply voltages are presented to ICs 1, 2 and 6. Turn off power.
3. Insert IC2 and apply power. The output at pin 2 should be 2.5 V and should not drift. Adjust the pot so that there is exactly 2.000 V on IC1 pin 2. Turn off power.
4. Insert the rest of the ICs including the MC14433. Be careful when inserting the 4051 and MC14433. You are now ready to wire the board to some convenient input and output ports and see if it flies.
5. Turn on power. A driver program obviously is necessary to see if the circuit actually works and I have included one. If you are really anxious, you can try a couple of quickies: an oscilloscope attached to digit select or data lines will tell you immediately if the circuit is running. You should see square waves of various duty cycles. Another method is to write a short program which scans the end of conversion bit (remember to reset it first) and halt. If it halts, there must be an EOC.

MSD

What You See is What You Get!

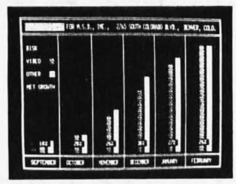
Now, A Video System You Can Afford!

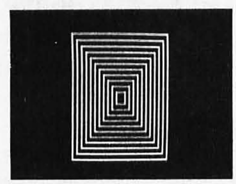
MSDV-100 Video Display System:

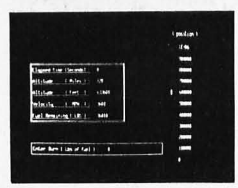
The Video Display System is a high quality 80 character, 24 line video output device for the S-100 bus. Many advanced features have been incorporated which are not normally found on units costing many times the price.

The character set includes upper and lower case characters as well as full punctuation. Any character can be underlined, a feature useful in word processing. A character can also be made to blink at a user selectable rate, often used for alarm or warning situations. Additionally a character can appear in reverse field (black on white) or, if composite video is used, individual characters can be intensified.

Also included in the MSDV-100 is the ability to generate high quality continuous forms overlays. Charts, graphs, or order entry forms are easy to produce on the video screen.







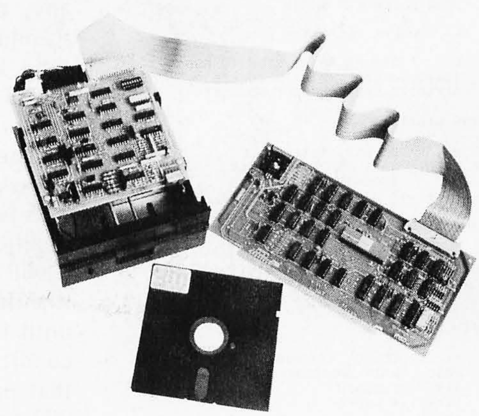
A third significant feature of the Video Display System is the ability to display grey scale elements in any of nine levels in any of 1920 positions on the screen. This is especially useful for bar graphs and for grey scale graphics or animations.

Internally, the MSDV-100 is a two-board S-100 based system which occupies 2K of RAM address space and two I/O ports, user selectable. The microcomputer can write to the screen directly with horizontal retrace synchronization if desired for a flicker free, very high speed display.

Software support for the MSDV-100 is complete with both machine language code, including fully commented source listings, and a comprehensive Basic software package implementing all MSDV-100 features. Assembly language drivers allow the sophisticated user to easily customize the system for specilized applications.

Programs are provided that permit the user to link the video system to high level programming languages such as Basic. A link program, provided in Basic, permits the user with no knowledge of assembly language programming to immediately obtain video output. The link fully implements the forms capability of the MSDV-100, including direct cursor addressing, as well as the other advanced features of the Video Display System.

MSDD-100 Floppy Disc System:



The MSDD-100 Floppy Disc System is a significant advance in low cost, high density mass storage systems. Using the industry standard Shugart SA400 minifloppy™ drive and a highly reliable LSI controller, the single card MSDD-100 Floppy Disc System represents a major cost/performance breakthrough for the hobbyist and businessman.

Many features not provided on larger disc systems are standard on the MSDD-100 Disc system. The controller will support up to three drives and provides all of the disc timing functions, therefore no software timing loops are required. A very flexible onboard vectored interrupt structure is provided, a valuable feature for use in modern multi-tasking applications.

The disc controller design is totally synchronous, requiring no "one shots". Ease of maintenance is evidenced by the fact that there are no adjustments required for operation.

The Altair/S100 compatible disc controller is a single board design, and features very low power consumption.

Included free with each MSDD-100 Floppy Disc System is a software package, provided on diskette, for formatting, certifying, and copying discs, as well as programs for creating fully customized memory-to-disc and disc-to-memory routines which may be put in read-only memory. In addition, assembly language I/O driver listings are provided to facilitate custom applications programming.

Also included are disc driver routines for Altair Basic, which allow program and data storeage on disc, and permit sector level I/O through Basic. Many programs and files may be kept on a single disc, and cassette I/O is retained. These drivers work with 8K, 3.2, 8K 4.0, Extended 3.2 and 4.1 versions of Basic.

MICRO SYSTEMS DEVELOPMENT

MSD INC

2765 So. Colorado Blvd. Suite 110 Denver, CO 80222 (303) 758-7411

Sanyo Monitor (VM4209)	**$150**
Micro-Floppy Disc System	**$499***
(Assembled)	**$599***
Video Display System	**$285**
(Assembled)	**$385**
Additional Drivers	**$350 ea.**
Diskettes	**$4.25 ea.**

**Power Supply not included.*

To place Order, send check, money order or BA or MC Card # with exp. date and signature. Uncertified checks require 6 weeks processing. Phone orders accepted.

Please Send me the following: **AMOUNT**

____

____

____

TOTAL: ______

Name __________

Address __________

City, State, Zip __________

☐ Send me more information

Circle 81 on inquiry card.

course) and called as a subroutine when the peripheral is to be exercised. The Digital Group Maxi BASIC, like many others, has instructions which allow memory and IO port manipulation as well as calling machine language subroutines. It is this latter call instruction which initiates the analog to digital conversion cycles and communicates with the interface driver program. When it executes this call instruction, it passes a channel convert code in the DE register pair. The driver program returns control to the BASIC interpreter at the conclusion of the analog to digital conversion. This provides a convenient method of synchronization. BASIC waits for the driver to finish storing the converted input data before trying to use it. Perhaps the next level is to write an interrupt driver which continually updates a value in the interpreter's tables of variables; but this would require a source listing and further documentation of the interpreter in order to accomplish the goal.

Listing 1, continued:

```
140224 113                    LD   C,E
140225                 *
140225                 * TEST POLARITY OF CHANNEL
140225                 *
140225 313 122         MSD1   BIT  2,D    TEST POLARITY
140227 040 017                JR   NZ,MSD2 JUMP IF POSITIVE
140231                 *
140231                 * NEGATIVE POLARITY
140231                 *
140231 014                    INC  C
140232 036 200                LD   E,200  LOAD NEGATIVE SIGN
140234 375 313 000 106        BIT  0,(IY+0) TEST PREVIOUS POLARITY
140240 040 022                JR   NZ,MSD3 JUMP IF ALSO NEGATIVE
140242 375 313 000 306        SET  0,(IY+0) MAKE PREVIOUS VALUE NEGATIVE
140246 030 314                JR   SCSC   CONVERT AGAIN
140250                 *
140250                 * POSITIVE POLARITY
140250                 *
140250 375 313 000 106 MSD2   BIT  0,(IY+0) TEST PREVIOUS POLARITY
140254 050 006                JR   Z,MSD3 JUMP IF ALSO POSITIVE
140256 375 313 000 206        RES  0,(IY+0) MAKE PREVIOUS VALUE POSITIVE
140262 030 300                JR   SCSC   CONVERT AGAIN
140264                 *
140264                 * SAVE MSD AND CURRENT POLARITY
140264                 *
140264 263             MSD3   OR   E      ADD POLARITY SIGN TO MSD
140265 335 167 000            LD   (IX+0),A SAVE IN DATA BUFFER
140270 375 161 000            LD   (IY+0),C SAVE CURRENT POLARITY
140273                 *
140273                 * PROCESS 2ND DIGIT
140273                 *
140273 313 010                RRC  B      SELECT DIGIT 2
140275 315 361 140            CALL RDIG   WAIT AND READ DIGIT
140300 346 017                AND  017    ISOLATE
140302 335 167 001            LD   (IX+1),A STORE SECOND DIGIT
140305                 *
140305                 * PROCESS 3RD DIGIT
140305                 *
140305 313 010                RRC  B      SELECT 3RD DIGIT
140307 315 361 140            CALL RDIG   WAIT AND READ DIGIT
140312 346 017                AND  017    ISOLATE
140314 335 167 002            LD   (IX+2),A STORE
140317                 *
140317                 * PROCESS 4TH DIGIT
140317                 *
140317 313 010                RRC  B      SELECT 4TH DIGIT
140321 315 361 140            CALL RDIG   WAIT AND READ DIGIT
140324 346 017                AND  017    ISOLATE
140326 335 167 003            LD   (IX+3),A STORE
140331 030 205                JR   INCCN
140333                 *
140333                 * LOAD 2.000 OVERRANGE VALUE INTO DATA BUFFER
140333                 *
140333 076 002         OVER   LD   A,2    LOAD MSD VALUE
140335 335 167 000            LD   (IX+0),A
140340 257                    XOR  A
140341 335 167 001            LD   (IX+1),A LOAD LSD VALUES
140344 335 167 002            LD   (IX+2),A
140347 335 167 003            LD   (IX+3),A
140352 303 140 140            JP   INCCN
140355                 *
140355                 * END OF CHANNEL CONVERSIONS
140355                 *
140355 052 044 140     RAPUP  LD   HL,(STATUS)
140360 311                    RET         RETURN TO CALLER
140361                 *
140361                 *
140361                 * READ DIGIT ROUTINE
140361                 *
140361 333 003         RDIG   IN   DIP    READ DATA BYTE
140363 057                    CPL         CONVERT TO HIGH TRUE LOGIC
140364 127                    LD   D,A    SAVE COPY
140365 240                    AND  B      TEST FOR GIVEN DIGIT READY
140366 050 371                JR   Z,RDIG JUMP IF NOT
140370 172                    LD   A,D    RESTORE A REGISTER
140371 311                    RET         RETURN TO CALLER
```

The Driver Is a Relocatable Subroutine

The actual program which interfaces to and stores the values to the DVM chip is written in the form of a single callable subroutine. To maintain the relocatability of the subroutine to any page in memory, all information necessary for the proper execution of the driver is provided at the time of the call. The additional information about which channels are to be converted is loaded into the DE registers at the time of the call. One bit of the E register is allocated for each analog to digital channel. Channel 1 is the least significant bit and channel 8 is the most significant. Setting a "1" value for the channel bit will tell the driver to convert that channel and a "0" means to ignore it. Loading E with binary 10 110 011 will indicate to the driver that channels 1, 2, 5, 6 and 8 are to be converted. Setting all bits to "1" will cause all channels to be read and converted. Indicating to the driver which, if any, channels are to be read rather than scanning all of them is a method of saving time. By computer standards, this analog to digital interface is slow; it is better not to waste any more time than is necessary.

The driver starts the conversion process by selecting a channel address to convert. This is accomplished by looking at the least significant bit of the E register. If it is a "1" it will convert on that channel. If it is a "0" it shifts and inspects the next bit, and so on until it finds one that is set. When a bit set condition is found, the channel address of that particular channel is sent out via port 003 to the analog input multiplexer IC6 and the end of conversion flip flop IC5 is reset. The DVM then starts the process of converting the analog input signal.

Demultiplexing the output of the DVM is fairly straightforward. The processor hangs in a loop waiting for an end of conversion signal. When this happens, the program knows that the next four digits of data are what is wanted. The DVM integrated circuit sets each of the digit select lines successively, and the program records the values of the four data lines each time. It strips the status and polarity bits from the most significant

Table 2: Power wiring table for figure 3.

IC Number	Type	+5V Pin	−5V Pin	GND Pin
IC1	MC14433	24	13	1&12
IC2	MC1403	1		3
IC3,4	74LS04	14		7
IC5	7474	14		7
IC6	CD4051	16	7	8

Note: All resistors ¼ W 5% unless otherwise noted.
All capacitors are 100 V ceramics unless otherwise noted.

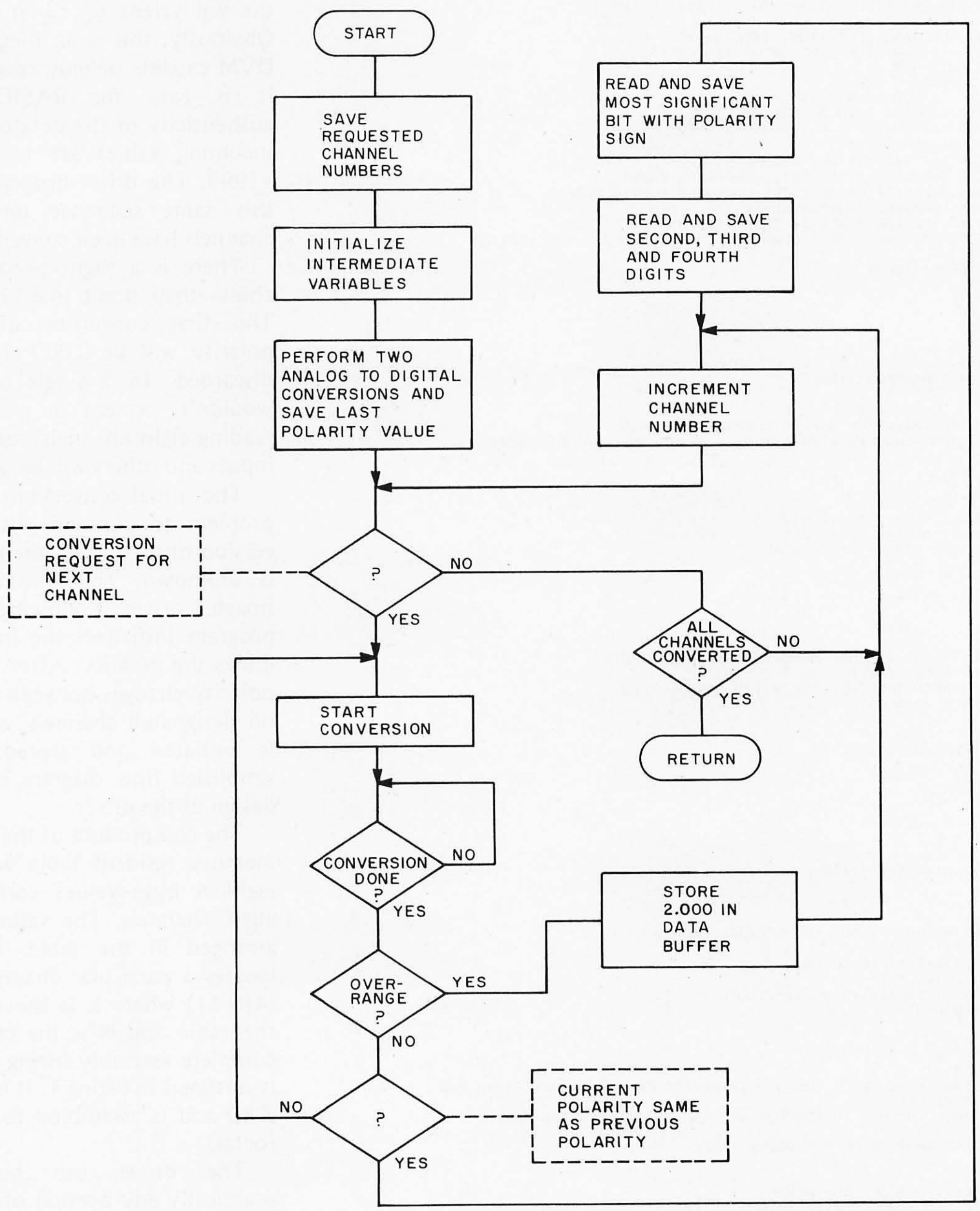


Figure 3: Flowchart of the digital voltmeter driver program of listing 1.

Listing 2: A BASIC program (written in Maxi BASIC) which performs data acquisition and computes results from the output of the 8 channel digital voltmeter.

```
LIST

100 REM
110 REM
120 REM 8 CHANNEL 3 1/2 DIGIT SCANNING PROGRAM -S.CIARCIA
130 REM REV 1.5
140 REM SPECIAL ANALYSIS SECTION ---
150 REM TTL TO MOS VOLTAGE LEVEL CONVERTER
160 REM
170 REM
180 REM
190 LET M1=24576
192 REM PAGE 140(OCTAL)
200 REM M1 IS SET TO BE THE DECIMAL STARTING LOCATION OF
210 REM THE VALUE TABLE
220 LET M2=24614
230 REM M2 IS THE MACHINE LANGUAGE CALL ADDRESS LOCATION FOR THE A/D
240 LET M3=10
250 REM M3 IS THE GAIN. IN THIS APPLICATION, THE RANGE OF THE CONVERTER
260 REM IS +19.99 TO -19.99 VOLTS
270 REM TO USE THE CONVERTER FOR -1.999 TO +1.999, LET M3=1
280 GOTO 300
290 PRINT"TO REPEAT THE SAME SELECTION, TYPE AN X"
291 PRINT"TO SELECT A NEW OPTION,TYPE AN O" :INPUT B$
292 IF B$="X" THEN GOTO 420
294 PRINT : PRINT : PRINT
300 PRINT"                    OPTION LIST"
310 REM WE START THE PROGRAM WITH AN OPTION LIST
320 PRINT"----------------------------------------------------"
330 PRINT"0 ----SELECT CHANNELS"
340 PRINT"1 ----SCAN AND DISPLAY ALL CHANNELS"
350 PRINT"2 ----SCAN AND DISPLAY SELECTED CHANNELS ONCE"
360 PRINT"3 ----SCAN AND DISPLAY SELECTED CHANNELS CONTINUOUSLY"
370 PRINT"4 ----SCAN CHANNEL 1 CONTINUOUSLY 100 TIMES"
380 PRINT"5 ----GO TO SPECIAL ANALYSIS SUBROUTINES"
390 REM THESE ROUTINES ARE DEPENDENT UPON THE PARTICULAR A/D APPLICATION
400 PRINT"6 ----EXIT"
410 PRINT"WHICH OPTION ":INPUT S
420 IF S=0 THEN 520
430 IF S=1 THEN 930
440 IF S=2 THEN 1060
450 IF S=3 THEN 1280
460 IF S=4 THEN 1380
470 IF S=5 THEN 1470
480 IF S=6 THEN PRINT"THANKYOU" :END
490 GOTO 410
500 REM
510 REM FIRST WE DETERMINE WHICH ANALOG CHANNELS TO READ
520 PRINT "INDICATE YOUR CHOICES WITH A Y OR N AFTER THE CHANNEL NUMBER
530 FOR C=1 TO 8
540 PRINT"CHANNEL ";C,
550 INPUT A$
560 REM ACCEPT ONLY TRUE INPUTS
570 IF A$="Y" THEN LET A(C)=1 :GOTO 610
580 IF A$="N" THEN LET A(C)=0 :GOTO 610
590 PRINT"INPUT A Y FOR YES OR A N FOR NO"
600 GOTO 540
610 NEXT C
620 GOTO 290
630 REM
640 REM
650 REM
660 REM SET D EQUAL TO THE DECIMAL MEMORY ADDRESS OF THE
670 REM BEGINNING OF VALUE TABLE
680 REM THIS SUBROUTINE DETERMINES THE 3 1/2 DIGIT VALUE
690 REM FROM THE TABLE IN MEMORY
700 LET Q1=EXAM(D)
710 LET Q=Q1
720 IF Q1>=128 THEN LET Q=Q1-128
730 D=D+1
740 LET W=EXAM(D)
750 D=D+1
760 LET E=EXAM(D)
770 D=D+1
780 LET R=EXAM(D)
790 LET D=D+1
800 LET Y=Q+(.1*W)+(.01*E)+(.001*R)
810 LET Y1=M3*Y
820 RETURN
830 REM
840 REM THIS SUBROUTINE PRINTS OUT THE VOLTAGE VALUES
850 PRINT"CHANNEL ";X;" IS ";
860 IF Q1<128 THEN PRINT" "; :GOTO 880
870 IF Q1>=128 THEN PRINT "-";
880 IF M3=10 THEN PRINT %5F2;Y1;" VOLTS" :GOTO 900
890 PRINT %6F3;Y1;" VOLTS"
900 RETURN
910 REM
920 REM
930 LET B=CALL(M2,255)
940 REM THE CALL INSTRUCTION TELLS THE A/D INTERFACE TO START CONVERTING
950 REM 255 IS ALL BITS SET
960 REM THIS WILL CAUSE THE A/D TO CONVERT AND STORE ALL EIGHT CHANNELS
970 LET D=M1
980 REM D IS THE START ADDRESS OF THE VALUE TABLE
990 FOR X=1 TO 8
1000 GOSUB 700
1010 REM GET 3 1/2 DIGIT VALUE FROM MEMORY
1020 IF Y>=2 THEN PRINT"CHANNEL ";X;" IS OUT OF RANGE" :GOTO 1040
1030 GOSUB 850
1040 NEXT X
1050 GOTO 290
```

digit (the 3½ digit) and reformats the value into four bytes of memory. The three whole digits will be stored in BCD notation and occupy three of the bytes. The ½ digit, polarity and out of range will be located in the remaining data byte. Polarity is indicated by setting the most significant bit. A positive reading is a zero condition and negative is a one in that bit. The ½ digit value can only be a one or zero and occupies the least significant bit of the quantity. Out of range is accomplished with a little program manipulation. If the driver detects that the incoming reading is not within range, it sets the equivalent of +2 in the ½ digit byte. Obviously, this is an illegal condition for a DVM capable of only counting to 1999, but it is easy for BASIC to check the authenticity of the data by checking that all incoming values are between −1999 and +1999. The driver program continues to do this same sequence until all designated channels have been converted.

There is a slight peculiarity with DVM chips: they don't like changes in polarity. The first conversion after a change in polarity will be 0.000 and will have to be discarded. In a single channel DVM this wouldn't present a problem, but when reading eight channels, some will be negative inputs and others will be positive.

The initial conversion also has the same problem to contend with, since the conversion history when the driver is not active is unknown. The solution is to write a smarter driver. Following a call, the driver program initializes the interface and determines the polarity. After that, any time the polarity changes between successive readings on designated channels, another conversion is initiated and stored. Figure 3 is a simplified flow diagram showing the logical design of the driver.

The end product of the driver is a 32 byte memory resident table which contains the eight 4 byte values corresponding to the eight channels. The values are sequentially arranged in the table. A simple formula locates a particular channel location at L + (4(N-1)) where L is the starting address of the table and N is the channel number. A complete assembly listing of the DVM driver is outlined in listing 1. It is made to run on a Z-80 and is assembled to occupy page 140 (octal).

The driver can be assembled for practically any portion of memory, but take care not to overlap into operating system or source files. If you own Digital Group soft-

Listing 2, continued:

```
1060 LET D=M1
1070 GOSUB 1130
1080 GOTO 290
1090 REM
1100 REM
1110 REM
1120 REM THIS SUBROUTINE PRINTS ONLY THE SELECTED CHANNELS
1130 LET L=A(1)*1+A(2)*2+A(3)*4+A(4)*8+A(5)*16+A(6)*32+A(7)*64+A(8)*128
1140 REM THIS EQUATION SETS THE BIT PATTERN FOR THE CALL TO THE A/D
1150 LET H=CALL(M2,L)
1160 REM H WILL RETURN FROM THE CALL WITH THE HL REG. VALUE BUT
1170 REM IS NOT BEING USED PRESENTLY IN THIS PROGRAM
1180 FOR X=1 TO 8
1190 GOSUB 700
1200 LET Z=A(1)+A(2)+A(3)+A(4)+A(5)+A(6)+A(7)+A(8)
1210 IF Z=0 THEN PRINT"NO CHANNELS HAVE BEEN SELECTED" :EXIT 290
1220 IF A(X)=0 THEN 1250
1230 IF Y>=2 THEN PRINT"CHANNEL ";X;" IS OUT OF RANGE" : GOTO 1250
1240 GOSUB 850
1250 NEXT X
1260 RETURN
1270 GOSUB 1130
1280 LET D=M1
1290 REM THIS SUBROUTINE IS A CONTINUOUS LOOP --- EXIT WITH RESET SWITCH
1300 FOR J=1 TO 1000
1310 LET D=M1
1320 GOSUB 1130
1330 PRINT : PRINT : PRINT
1340 NEXT J
1350 GOTO 290
1360 REM
1370 REM THIS SUBROUTINE CONTINUOUSLY SCANS AND PRINTS CHANNEL 1
1380 FOR R=1 TO 100
1390 LET H=CALL(M2,1)
1400 REM GO AND CONVERT CHANNEL 1 ONLY
1410 LET D=M1
1420 GOSUB 700
1430 LET X=1
1440 GOSUB 850
1450 NEXT R
1460 GOTO 290
1470 LET B=CALL(M2,255)
1480 REM SCAN AND STORE ALL CHANNELS
1490 LET D=M1
1500 FOR X=1 TO 8
1510 GOSUB 700
1520 LET V(X)=Y1
1530 REM SET VALUES INTO AN 8 VALUE ARRAY
1540 NEXT X
1550 REM CHECK CALIBRATION
1560 IF V(8)>=2.006 THEN PRINT"OUT OF CALIBRATION" :GOTO 290
1570 IF V(8)<=1.994 THEN PRINT"OUT OF CALIBRATION" :GOTO 290
1580 PRINT"TTL TO MOS LEVEL CONVERTER ---TTL LOW INPUT STATE":PRINT:PRINT
1590 PRINT"DIODE D1"
1600 PRINT"VOLTAGE DROP =",V(2)-V(1);" VOLTS"
1610 PRINT :PRINT"R1"
1620 LET T1=(V(2)-V(3))/2200
1630 PRINT"CURRENT = ";T1;" AMPS"
1640 PRINT"POWER = ";T1*T1*2200
1650 PRINT :PRINT"Q1"
1660 PRINT"VCE Q1 = ";V(4)-V(3);" VOLTS"
1670 PRINT"VBE Q1 = ";V(3);" VOLTS"
1680 PRINT"VCB Q1 = ";V(4);" VOLTS"
1690 PRINT :PRINT"R2"
1700 LET T2=(V(5)-V(4))/4700
1710 PRINT"R2 DROP = ";V(5)-V(4);" VOLTS"
1720 PRINT"CURRENT = ";T2;" AMPS"
1730 PRINT"POWER = ";T2*T2*4700;" WATTS"
1740 PRINT :PRINT"R3"
1750 LET T3=(V(6)-V(7))/4700
1760 PRINT"R3 DROP = ";V(6)-V(7);" VOLTS"
1770 PRINT"CURRENT = ";T3;" AMPS"
1780 PRINT"POWER = ";T3*T3*4700;" WATTS"
1790 PRINT:PRINT"Q2"
1800 PRINT"VBE Q2 = ";V(5)-V(4);" VOLTS"
1810 PRINT"VCE Q2 = ";V(7)-V(5);" VOLTS"
1820 PRINT"VCB Q2 = ";V(7)-V(4);" VOLTS"
1830 PRINT"IC OF Q2 = ";T3;" AMPS"
1840 PRINT"POWER DISSIPATION = ";(V(7)-V(5))*T3;" WATTS"
1850 PRINT:PRINT:PRINT"SUPPLY VOLTAGES"
1860 PRINT V(6)
1870 PRINT"-";V(5)
1880 GOTO 290
READY


RUN

                 OPTION LIST
------------------------------------------------------------
0 ----SELECT CHANNELS
1 ----SCAN AND DISPLAY ALL CHANNELS
2 ----SCAN AND DISPLAY SELECTED CHANNELS ONCE
3 ----SCAN AND DISPLAY SELECTED CHANNELS CONTINUOUSLY
4 ----SCAN CHANNEL 1 CONTINUOUSLY 100 TIMES
5 ----GO TO SPECIAL ANALYSIS SUBROUTINES
6 ----EXIT
WHICH OPTION
?1
CHANNEL  1 IS   3.54 VOLTS
CHANNEL  2 IS   2.93 VOLTS
CHANNEL  3 IS    .63 VOLTS
CHANNEL  4 IS -10.75 VOLTS
CHANNEL  5 IS -11.02 VOLTS
CHANNEL  6 IS   5.04 VOLTS
CHANNEL  7 IS -11.45 VOLTS
CHANNEL  8 IS   2.00 VOLTS
TO REPEAT THE SAME SELECTION, TYPE AN X
TO SELECT A NEW OPTION,TYPE AN O
?O

                 OPTION LIST
------------------------------------------------------------
0 ----SELECT CHANNELS
1 ----SCAN AND DISPLAY ALL CHANNELS
2 ----SCAN AND DISPLAY SELECTED CHANNELS ONCE
3 ----SCAN AND DISPLAY SELECTED CHANNELS CONTINUOUSLY
4 ----SCAN CHANNEL 1 CONTINUOUSLY 100 TIMES
5 ----GO TO SPECIAL ANALYSIS SUBROUTINES
6 ----EXIT
WHICH OPTION
?0
INDICATE YOUR CHOICES WITH A Y OR N AFTER THE CHANNEL NUMBER
CHANNEL  1     ?Y
CHANNEL  2     ?N
CHANNEL  3     ?N
CHANNEL  4     ?Y
CHANNEL  5     ?Y
CHANNEL  6     ?N
CHANNEL  7     ?N
CHANNEL  8     ?N
TO REPEAT THE SAME SELECTION, TYPE AN X
TO SELECT A NEW OPTION,TYPE AN O
?O

                 OPTION LIST
------------------------------------------------------------
0 ----SELECT CHANNELS
1 ----SCAN AND DISPLAY ALL CHANNELS
2 ----SCAN AND DISPLAY SELECTED CHANNELS ONCE
3 ----SCAN AND DISPLAY SELECTED CHANNELS CONTINUOUSLY
4 ----SCAN CHANNEL 1 CONTINUOUSLY 100 TIMES
5 ----GO TO SPECIAL ANALYSIS SUBROUTINES
6 ----EXIT
WHICH OPTION
?2
CHANNEL  1 IS   3.54 VOLTS
CHANNEL  4 IS -10.75 VOLTS
CHANNEL  5 IS -11.02 VOLTS
TO REPEAT THE SAME SELECTION, TYPE AN X
TO SELECT A NEW OPTION,TYPE AN O
?O

                 OPTION LIST
------------------------------------------------------------
0 ----SELECT CHANNELS
1 ----SCAN AND DISPLAY ALL CHANNELS
2 ----SCAN AND DISPLAY SELECTED CHANNELS ONCE
3 ----SCAN AND DISPLAY SELECTED CHANNELS CONTINUOUSLY
4 ----SCAN CHANNEL 1 CONTINUOUSLY 100 TIMES
5 ----GO TO SPECIAL ANALYSIS SUBROUTINES
6 ----EXIT
WHICH OPTION
?5
TTL TO MOS LEVEL CONVERTER ---TTL LOW INPUT STATE

DIODE D1
VOLTAGE DROP =        -.61 VOLTS
R1
CURRENT =  1.0454545E-03 AMPS
POWER =  2.4045452E-03
Q1
VCE Q1 =  10.12 VOLTS
VBE Q1 =  .63 VOLTS
VCB Q1 =  10.75 VOLTS
R2
R2 DROP =  .27 VOLTS
CURRENT =  5.7446809E-05 AMPS
POWER =  1.5510639E-05 WATTS
R3
R3 DROP =  -6.41 VOLTS
CURRENT =  -1.3638298E-03 AMPS
POWER =  8.742149E-03 WATTS
Q2
VBE Q2 =  .27 VOLTS
VCE Q2 =  .43 VOLTS
VCB Q2 =  .7 VOLTS
IC OF Q2 =  -1.3638298E-03 AMPS
POWER DISSIPATION =  -5.8644681E-04 WATTS
SUPPLY VOLTAGES
 5.04
- 11.02
TO REPEAT THE SAME SELECTION, TYPE AN X
TO SELECT A NEW OPTION,TYPE AN O
?O

                 OPTION LIST
------------------------------------------------------------
0 ----SELECT CHANNELS
1 ----SCAN AND DISPLAY ALL CHANNELS
2 ----SCAN AND DISPLAY SELECTED CHANNELS ONCE
3 ----SCAN AND DISPLAY SELECTED CHANNELS CONTINUOUSLY
4 ----SCAN CHANNEL 1 CONTINUOUSLY 100 TIMES
5 ----GO TO SPECIAL ANALYSIS SUBROUTINES
6 ----EXIT
WHICH OPTION
?6
THANKYOU
READY
```

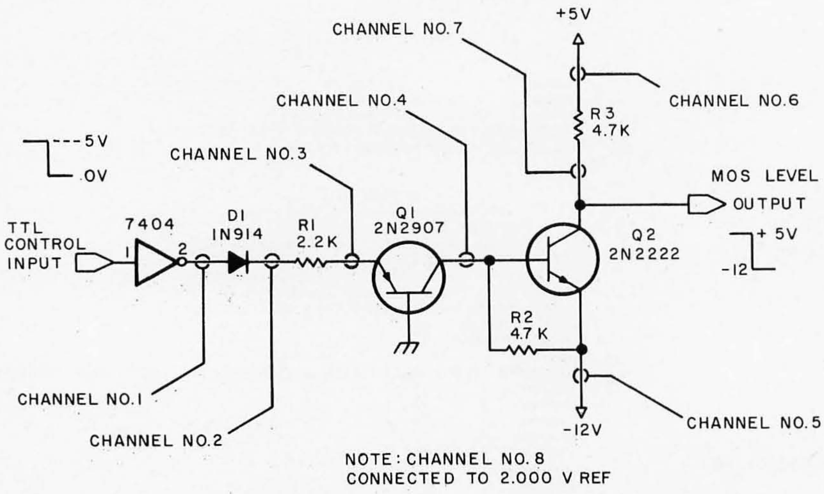


Figure 4: A sample circuit illustrating the use of the 8 channel 3½ digit voltmeter. The circuit is a TTL to MOS voltage level converter.

ware, there are some alternatives depending on what version you have. For people with straight (non-universal) 32 character Z-80 Maxi BASIC Version 1.0, page 012 is empty and has been left for future expansion. If you have the 64 character Maxi BASIC Version 1.1, it's better not to try to bury the driver within the interpreter unless you're an experienced programmer. Owners of 8080 systems have only to reassemble the code using 8080 instructions and locate it in a similar manner. The logic behind the driver is not so involved that it necessitates using the Z-80. Any microprocessor should be able to work with the interface.

Using the Interface with BASIC

This DVM interface is specifically designed to run with a BASIC interpreter such as Maxi BASIC or the equivalent. Listing 2 illustrates a BASIC program which does data acquisition and computes results from this input data. Often, the best method of explanation is to illustrate the actual use of a device. This program, while being general in nature, provides specific reference to the value of mating BASIC and analog acquisition.

Figure 4 is a circuit of a TTL to MOS voltage level converter. Its use is to convert 0 and 5 V TTL levels to +5 V and −12 V MOS logic levels. It is a relatively simple circuit, but it shows how BASIC can work for you.

Up to this point I have said that the input range of the DVM is ±1.999 V. By putting resistor voltage dividers in series with the multiplexer channel inputs, other ranges can be accommodated. A 900 K-100 K resistor divider network will change the input range to ±19.999 V. Some channels can be set for 20 V ranges. With the present CD4051, though, separate resistor dividers are needed on the inputs because the maximum voltage handling capability of the 4051 is the range of its power supply. Relays, which could pass the high voltages, could be configured to allow use of only one selectable divider network, but for now we are limited. If you put resistor dividers on the inputs, the only necessity is to instruct the program to multiply the particular channel reading by an appropriate ranging factor. In this particular case, all input channels have been set for ±19.99 V ranges, and the multiplier is ten.

The program presents an option list. It allows general application as an acquisition and data logging tool. With it, one can select to read and print all eight channels, particular channels, or log a single channel continuously. Option 5 is what it's all about. It automatically records the input voltages and computes the circuit parameters such as power dissipation and voltage drops. A very complicated circuit example would probably have been more impressive, but that is merely a case of applying programming talents to the same set of input data.

One further note of explanation: the call instruction in Maxi BASIC has been misinterpreted by some people. It is not a directly executable instruction, but is rather used in a statement like LET X = CALL (2560,9). The BASIC interpreter will go to memory location decimal 2560 and start executing a machine language subroutine. The number in parentheses after the comma is the value which is put in the D and E registers at the same time. This is a 16 bit value with a range of 0 to 65,535. When the machine language subroutine is finished, it returns to the interpreter. X will then have a value equal to whatever was in the H and L registers when the subroutine ended.

Conclusion

Having eight channels is better than having one, especially if it doesn't cost any more. I've attempted to present a low cost solution to a usually expensive data acquisition problem. As is always the case with computers, the maximum utilization of the device is dependent upon the programmer, and as my college textbooks used to say, this is an exercise left to the reader.

If you have a suggestion for an article or an idea about a project to be built, please write and tell me about it. Please enclose a stamped, self-addressed envelope. Unfortunately, based on the mail volume of previous articles, I will not be able to answer all letters personally, but I will attempt to.■

The author would like to extend special thanks to Dave Hardenbrook for his help in writing the DVM driver program.

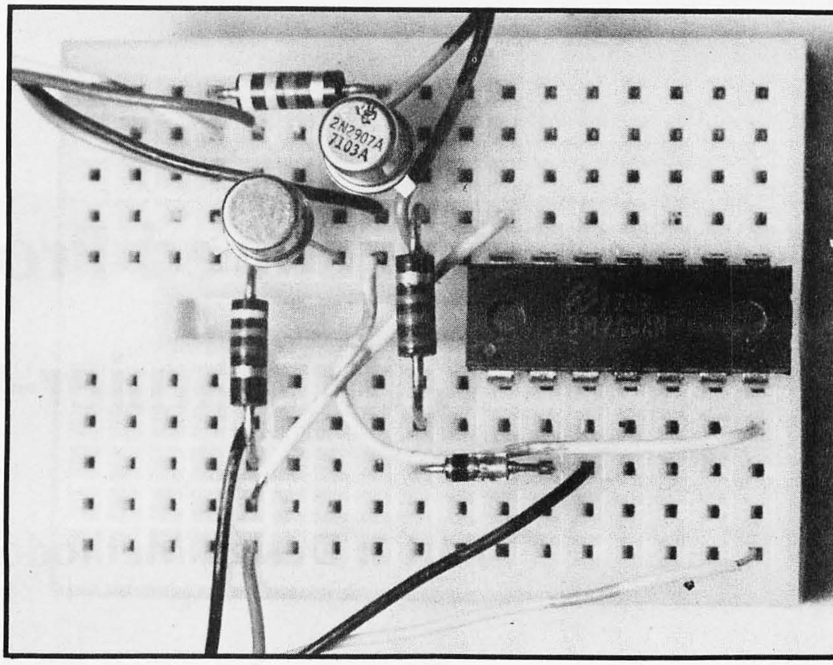


Photo 5: The breadboard circuit of the schematic in figure 4 used to test the 8 channel voltmeter.

Structured Programming with Warnier-Orr Diagrams

Part 1: Design Methodology

David A Higgins
Langston Kitch and Associates
715 E 8th St
Topeka KS 66607

Any successful program design methodology must be able to do several things: it must produce consistent, low cost, high reliability results; it must produce them quickly, while still allowing for easy maintenance later and, it must be simple enough to allow anyone (and I do mean anyone) to use it. Warnier-Orr diagrams (after Jean-Dominique Warnier in France and Kenneth T Orr in the United States) satisfy all of the above requirements with an added bonus; they produce structured programs that nearly always run correctly at the first effective trial. They allow people to produce superprograms without being superprogrammers.

The purpose of this article is to show how to develop and code a structured program using the Warnier-Orr methodology from start to finish. The technique is a straightforward approach to producing correct programs. It is just as valid and successful for personal microcomputer applications as it is for megacomputer applications in the world of business, science and industry. I feel that this method of designing a program is one of the most advanced state of the art software development techniques in existence today. It is a concise, step by step method with predictable results.

Step One: Identify the Output

This is the first, the primary and the most important rule of all for the construction of a correct program. It cannot be emphasized enough. The failure to **first** identify the outputs of a program is usually the primary reason programs fail to run correctly.

You must ask yourself the questions: "How will I be able to tell when I am through with this program?" "What will the printed, displayed and punched outputs physically look like?" "What will the program be able to do?" All of these questions must be thoroughly answered before you can even begin to think of coding the program. Skipping this step because "Aw, I know what I want to do," or "Gee, this isn't any fun, let's start coding," is a common mistake, and although you may get away with it on a small program once in a while, omitting it will kill you more often than not.

A good example of the kind of trouble you can get into by assuming that you know everything about a problem can be found in a recent popular film. In the movie *Jeremiah Johnson*, Jeremiah befriends an old·hunter and trapper in the mountains. The old hunter asks Jeremiah if he can skin a bear. "Of course I can," he replies. In the next scene, we see the old man running down a hill towards the cabin closely pursued by a very large bear. The hunter runs into the open front door, leaps out of the back window and yells: "There . . . you skin that one and I'll go get you another." Jeremiah failed to do one basic thing; he forgot to ask whether the bear he was supposed to skin was dead. Skinning a dead bear is one thing, skinning one that is still running around the room trying to skin you is quite another. Just as writing a program after it has been properly defined is one thing, and trying to write one when you aren't even sure what it is supposed to do when you are finished is another.

Defining outputs is not really an unreasonable requirement to make; after all, no building contractor would begin construction without first knowing what the finished building was supposed to look like; no electrical engineer would start soldering

YES YOU CAN

Put The VERSATILE 2 To Work Now!

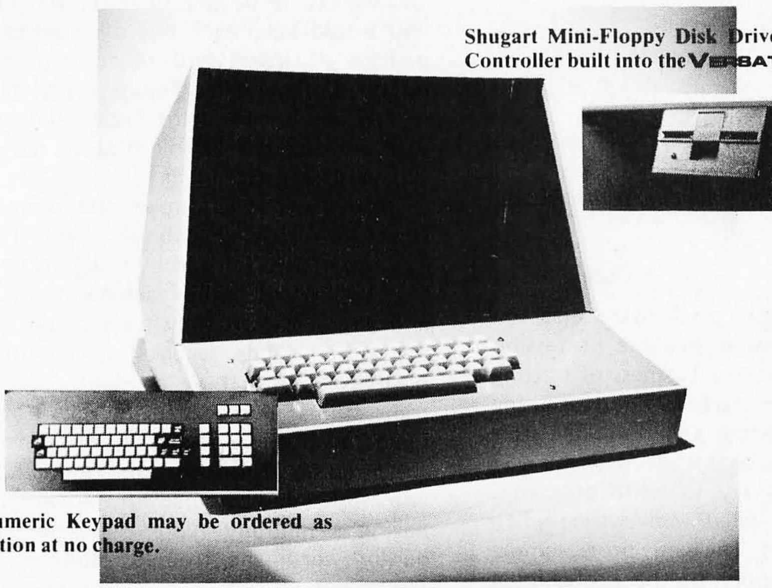
Even if you've NEVER used a computer, you can begin to operate the VERSATILE 2 immediately.
It's all been done for you: a compact computer that needs no extras, and a complete software library.

Great for HOBBYISTS • TEACHERS • BUSINESSMEN

HOME You don't have to be a programmer or electronics technician. Or know anything more than how to plug it in, and read our manual. Get your hobby going without hassle.

CLASSROOM Get your kids right to work learning BASIC, playing games and solving math and accounting problems without worrying about what makes the computer go.

OFFICE Stop drowning in paperwork. And quit worrying about which system will work best for you. With our business accounting package, you can concentrate on business while the VERSATILE 2 concentrates on maintaining your records compactly, and permanently on diskette.



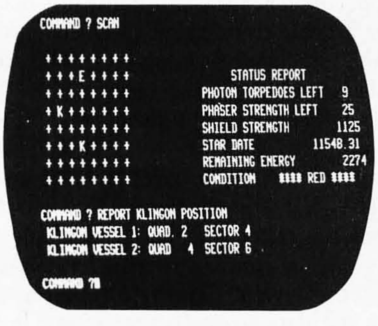
Shugart Mini-Floppy Disk Drive and Controller built into the VERSATILE 2

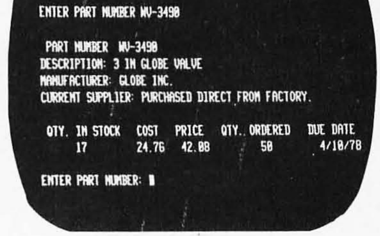
Numeric Keypad may be ordered as option at no charge.

SPECIFICATIONS
Z-80 CPU
Video display with graphics on 9" 64x16 screen
16K Static RAM Memory
Serial and Parallel I/O Ports with standard RS-232 Connector provided at rear of unit
ROM to drive RS-232

EXPANDABLE
Add memory, printer, and up to 3 external mini-floppy disk drives.

OPTIONS
8K Memory Boards at $195.00 each.





HOME SOFTWARE LIBRARY

Five diskettes are included to give you immediate programming capabilities.

DISK #1 contains a Disk Operating System and 12K Extended BASIC. Easy to use statements include: IF THEN, GOTO, READ, EXIT, FOR, NEXT. You get complete line editor, multi-statement lines and multi-dimension arrays. A BASIC teaching program on this diskette will have you quickly programming in BASIC.

DISK #2 has many games including STARTREK, BLACKJACK and STAR WARS. There's room left over for you to add your own.

DISK #3 is a Home Accounting Package with programs like Budgeting, and Checkbook Balancing.

DISK #4 contains a Small Business Accounting Package. Included are programs for Payroll, Inventory Control, Accounts Payable and Receivable, Taxes, Invoicing, Check Printing, and much more.

DISK #5 is a formatted blank diskette for you to enter your own programs.

VERSATILE 2

$2495 Assembled and Tested. 6 Month Warranty. 30-day delivery or available direct from dealers.

COMPUTER DATA SYSTEMS
5460 Fairmont Drive
Wilmington, Delaware 19808
(302) 738-0933

Scientific Sales, Inc.
175 W. Wieuca, Suite 210
Atlanta, GA 30342
(404) 252-6808

Alexander and Company, Inc.
5518 Florin Road
Sacramento, CA 95823
(916) 422-9070

Circle 25 on inquiry card.

```
                    MONTHLY FINANCIAL REPORT
                  FOR THE MONTH OF JANUARY 1977

                                      BALANCE FORWARD OF   $231.90

DATE    CHECK#    TO:                 DEBIT     CREDIT     BALANCE
 1       978       GROCERY STORE       2.23                 229.67
                   -MILK, BREAD, EGGS

 1       979       PHONE COMPANY      37.14                 192.53

 3       980       GAS BILL           25.61                 166.92

 5       981       GEORGE FREDRICK     5.00                 156.92
                   -SHOVELLING SNOW

 5                 PAYCHECK DEPOSIT             312.18      469.10

 6       982       ELECTRIC COMPANY   23.15                 445.95
           .                            .                     .
           .                            .                     .
           .                            .                     .
31      1013       BYTE MAGAZINE      12.00                 237.11
                   -SUBSCRIPTION RENEWAL

                                      CURRENT BALANCE       237.11
```

Figure 1: Proposed output of a computer program for balancing a checkbook and producing an end of month report.

parts together without a schematic diagram. In fact, no profession (reliable profession anyway) involved in the business of putting things together ever starts to build anything unless they know what it will look like after they are done. Yet, that is precisely the way most programmers try to write programs. Then they wonder what went wrong when they have problems. The same programming principles which apply to the professional apply just as much to the amateur, for no one's time is unlimited.

After defining all of the outputs of the program, the next step is to define the logical data base, although you will probably never really spend much time at this step with most personal microcomputer applications.

Step Two: Define the Logical Data Base

The reason this step is trivial for many personal use applications is because the logical data base typically consists of only one numeric field. It is typically the field holding a person's response to a program generated question. For illustrative purposes let us look at a home computer application that requires a slightly more complex data base arrangement. Take for instance a computer program that would balance the family checkbook and produce a financial report each month. The report designed in step one might look something like figure 1.

If you were keeping manual records that you wanted to be able to search very easily, you would keep each one of those entries, perhaps on index cards, filed by year, by month and by date. Figure 2 illustrates a way of representing the logical data structure for the checkbook balance report in Warnier-Orr notation.

In figure 2, you can see the logical data structure for the checkbook balance report. The report is organized by year; within each year by months; within each month by days; and within each day by transactions, which are either debits (checks) or credits (deposits). Note that year, month, day, and transactions all appear in the report at least once and possibly many times; thus we see the notation (1,n) in the diagram. Having an entry for a day that had no transactions or having a monthly report with no days is hardly worth the trouble. However, each transaction is either a credit transaction (credit occurring once, and debit not occurring) or a debit transaction (debit occurring once and credit not occurring). This condition is reflected on the chart by the "⊕" symbol, which is the symbol for mutual exclusion.

One important point needs to be made here. The diagram of figure 2 is *not* the logical data base for this report; it is only the report's logical data structure. Making a chart of the logical data base requires that we map the data elements that appear in the report onto the logical report structure, as we have done in figure 3. In figure 2 we showed conceptual relationships of one part of the structure to another. In figure 3 we've filled in the required details needed to complete each level of the structure. One level of the structure corresponds to one bracket and the levels are counted left to right.

Step Three: Define the Physical Data Base

Defining the physical data base of a program is largely a packaging decision: what physical arrangement of the data in the

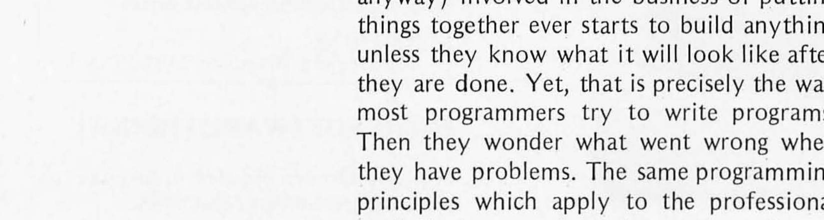


Figure 2: Logical data structure for the checkbook balance report. The notation (1,n) indicates an operation will take place at least once and possibly many times.

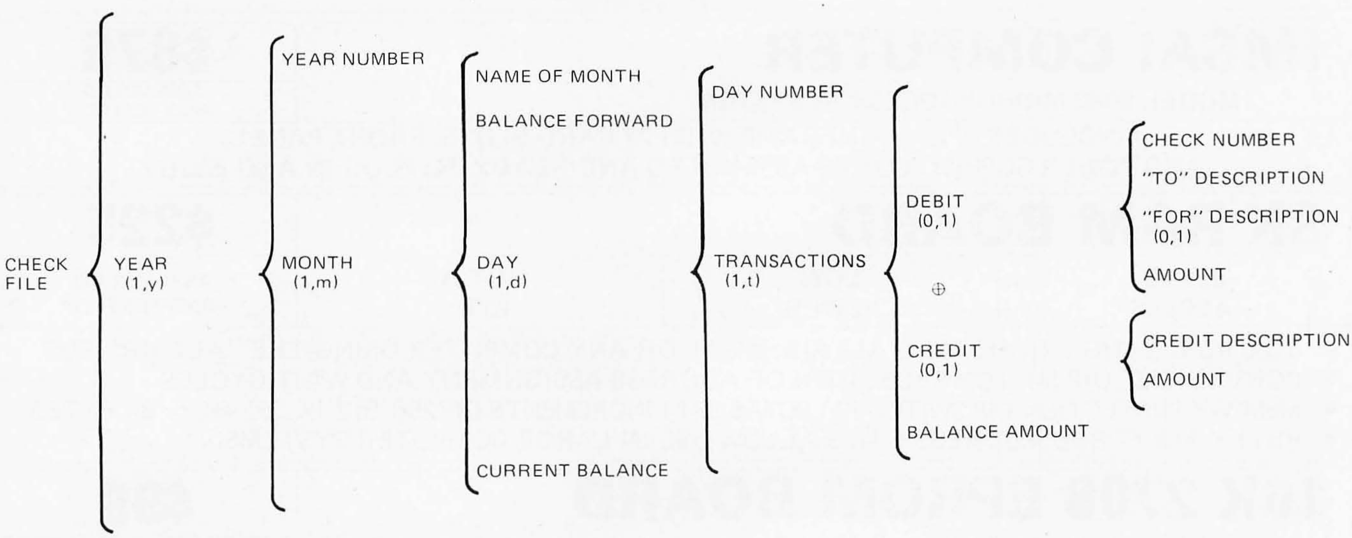


Figure 3: The logical data base is generated by mapping the data elements that appear in the report onto the logical data structure.

computer will best suit the needs of the program. The only help I can give you on this is the simple suggestion that the physical representation should mirror the logical representation in all but the most extreme cases. These are hardware decisions. You may wish to construct a file one way if you are using a cassette tape storage system; you may construct it another way if you have a floppy disk. You would not want to impose a file structure that forced a cassette tape to behave like a disk by running back and forth through the tape at high speed. That is a good way to burn up a tape drive in a hurry. Ultimately, as memories become faster, more versatile and more efficient, the physical data base will probably always be able to mirror the logical data base. Magnetic bubble memories, for instance, have no moving parts to burn up.

In the checkbook balance report program the simplest physical data base would be a sequential file. The necessary information and a brief description of each transaction could be stored in the order shown in figure 4, read left to right.

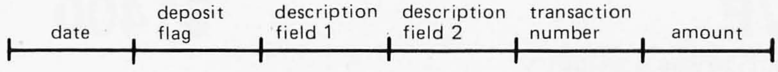


Figure 4: A sequential file with a record format such as this is the simplest physical data base for the checkbook program. The information that is needed has been decided by the logical data base. The order they are put on the file depends on exactly what you intend to do. Since in this case we will be sorting by date, the date of the transaction appears first on the file.

Given that we have a file with this information on it which is sorted by year, month, day and transaction, producing a report program is almost a trivial exercise.

Step Four: Design the Process Structure

Since in this case we are working with a single program, the process structure will ultimately represent the program structure. Were we designing an entire system, an accounts receivable system for instance, the process structure would represent many programs and the associated system procedures that would operate them. The process structure is obtained from the same logical data structure that the logical data base was derived from.

Referring again to both figures 1 and 2, we can begin to design the program from the bottom to the top. Looking first at the leftmost bracket, which for this step is labeled REPORT PROGRAM, we could draw a structure thus:



Note that program structure is denoted by left to right positioning, and that sequences of operations are noted top (first) to bottom (last).

We can see that the only thing for us to do at the beginning of the program is to open the files, and the only thing to do at

the end of the program is to close the files we have used. Moving right to the YEAR bracket, the process END YEAR must be defined. For this program there is nothing to do at the end of the year, so we fill in the bracket with the notation SKIP.

- YEAR (1,y)
  - END YEAR
    - SKIP

For the bracket labeled MONTH, there is the matter of printing the CURRENT BALANCE at the end of the month.

- MONTH (1,m)
  - END MONTH
    - PRINT CURRENT BALANCE

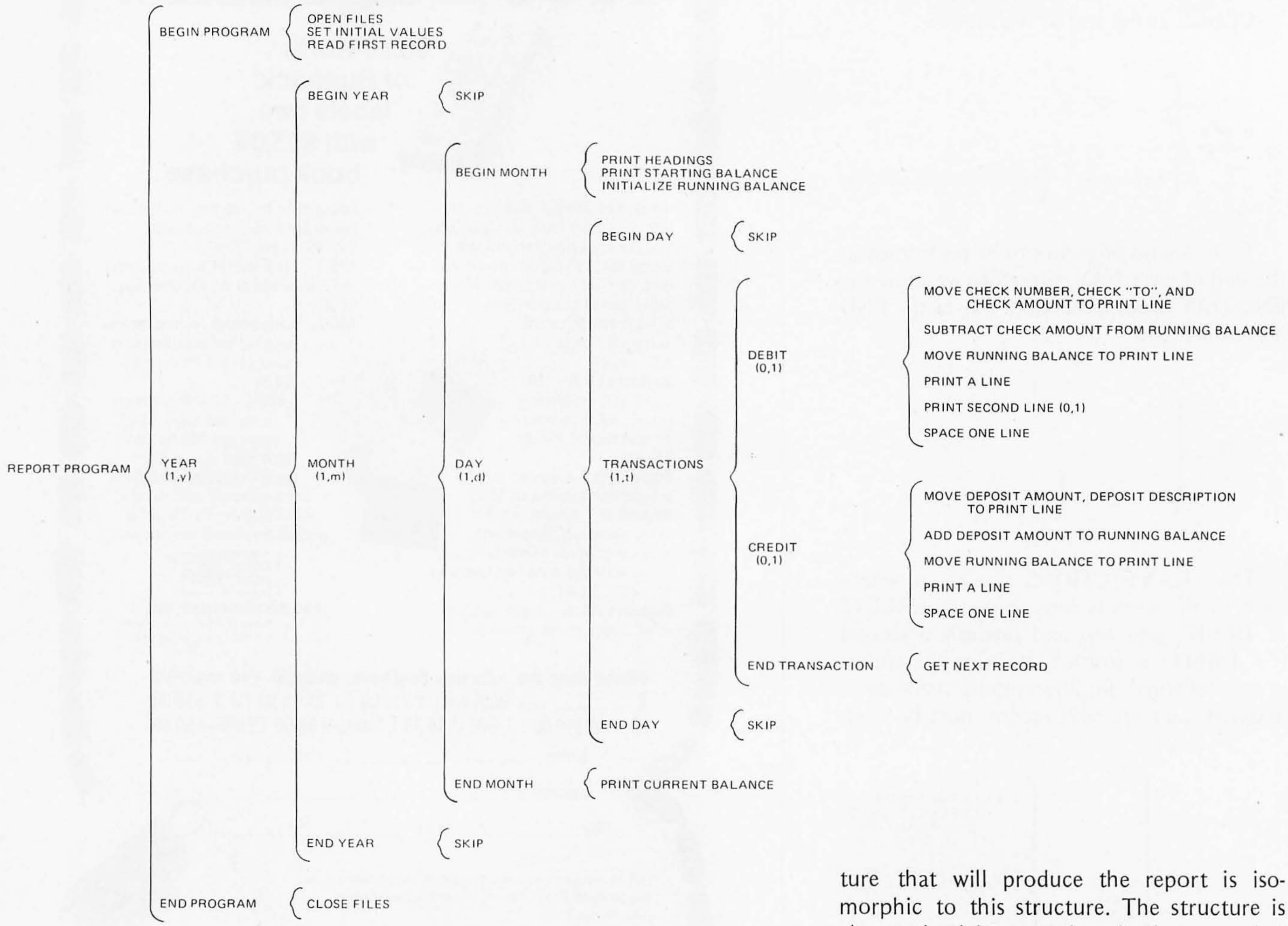
There are no processes to be performed at the end of each DAY, therefore we show the END DAY process the same way as the END YEAR process.

- DAY (1,d)
  - END DAY
    - SKIP

The TRANSACTIONS process is where most of the work is done. For each CREDIT or DEBIT, one line and possibly a second (for DEBIT) is printed, showing the appropriate information; the running balance is updated, and the next record must be read.

- TRANSACTIONS (1, t)
  - DEBIT (0,1)
    - MOVE CHECK NUMBER, CHECK "TO", AND CHECK AMOUNT TO PRINT LINE
    - SUBTRACT CHECK AMOUNT FROM RUNNING BALANCE
    - MOVE RUNNING BALANCE TO PRINT LINE
    - PRINT A LINE
    - PRINT SECOND LINE (0,1)
    - SPACE ONE LINE
  - ⊕
  - CREDIT (0,1)
    - MOVE DEPOSIT AMOUNT, DEPOSIT DESCRIPTION TO PRINT LINE
    - ADD DEPOSIT AMOUNT TO RUNNING BALANCE
    - MOVE RUNNING BALANCE TO PRINT LINE
    - PRINT A LINE
    - SPACE ONE LINE
  - END TRANS-ACTION
    - GET NEXT RECORD

Figure 5: Completed Warnier-Orr diagram for a checkbook balancing report program. This program arrangement will probably result in the smallest amount of memory being used. The sequences of operations at any given level (left-right position) are read from top to bottom. A level of operations corresponds to a logical level of procedure calls in a block structured programming language.

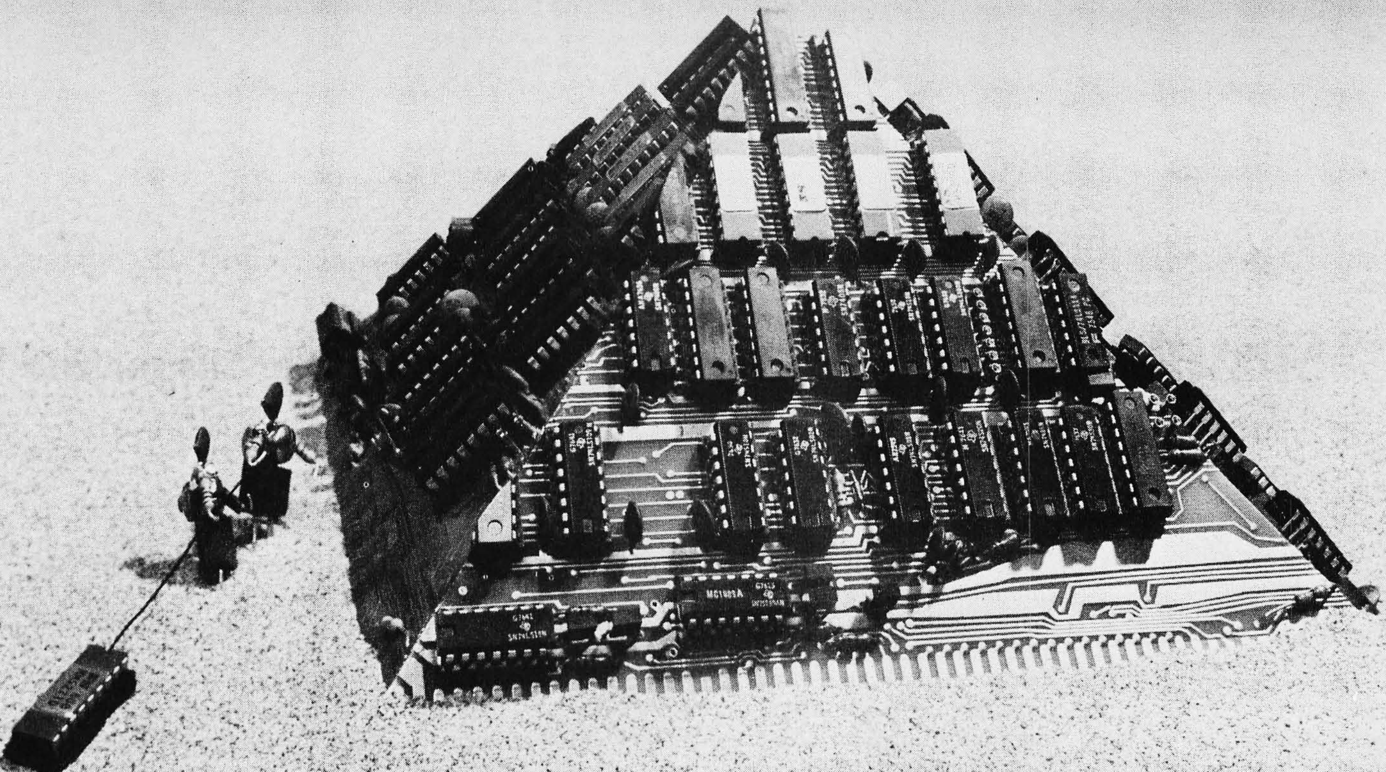


With this much of the program design done, the only things to be filled in are the BEGIN brackets for each level. The entire diagram with these processes added is shown in figure 5.

Looking at the Warnier-Orr diagram for the checkbook balance program, you can see the entire series of events which must take place to correctly process the report as it was given. Note also that this is the only correct structure that will produce the checkbook balance report. Any other structure that will produce the report is isomorphic to this structure. The structure is also optimal in operation, in the sense that nothing is ever done unless it must be done.

The program which is coded from this structure will also have some predictable features. It will run as quickly as possible. It will usually require the least amount of storage. It is very easy to maintain, and it will run correctly at the first effective trial. Not bad dividends for a half hour of extra work. Syntax runs are not effective trials, but, with a little diligence and effort, syntax errors can also be brought under control.

Next month Part 2 will show how easy it is to fill in the details of structured programs using Warnier-Orr diagrams.■



Dynabyte builds the Great Memory

We cut up a Dynabyte 16k dynamic RAM board and constructed this pyramid to illustrate an important point: Dynabyte designs and builds memory boards with the same unmatched engineering ability and technical skill that went into Egypt's Great Pyramid.

One of the seven wonders of the ancient world, the Great Pyramid has been standing on the desert for an incredible 4,400 years. Although its enormous base covers 13 acres, it is perfectly square. Rising 450 feet, it is as tall as a 37 story building. Over 2.3 million blocks of stone were used, each averaging 2½ tons. Some weigh 16 tons. Despite their size, they fit together with a tolerance that is less than half the width of a human hair.

Dynabyte builds its 16k dynamic RAM boards with the same exceptional precision and care. Their reliability is as solid as a rock.

Dynabyte's design meets rigid industrial grade standards. The design is so good, in fact, that one of the largest, most experienced electronics manufacturers has tried to imitate it. (We were flattered but not surprised; we know how good it is.)

More than 1400 microcomputer owners also know how good it is. Dynabyte's 16k dynamic is running in more systems than any other dynamic memory on the market.

We select the best components we can buy to build the 16k dynamic, because solid parts make a solid memory. Our memory chips, for example, are factory prime from National Semiconductor.

Dynabyte was the first to deliver 16k dynamic RAM's assembled, tested and burned in. And at a price competitive with kits! Each board's complete function is confirmed by three stages of testing and a burn in cycle that runs 72 hours at 70° C (158°).

When we build them that solid we can guarantee them for a full year.

If a Dynabyte board ever needs repair, we provide factory service with a 24 hour turnaround for both warranty and non-warranty work.

The Dynabyte 16k dynamic has the widest compatibility of any dynamic memory. So it will work in your system.

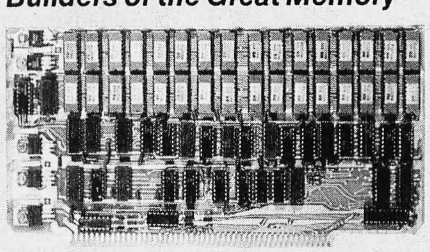
The Great Memory by Dynabyte is a solid buy. And an economical one. Effective October 1, the new Manufacturer's Suggested Price is reduced from $485 to $399.

Ask for the Great Memory by Dynabyte at your local computer store. If it isn't in stock, tell the owner that he missed another Dynabyte sale, and order direct. Telephone (415)494-7817. Cable DYNABYTE. Or mail to Dynabyte, Inc., 4020 Fabian, Palo Alto, CA 94303.

Specifications: 16,384 bytes, National Semiconductor MM5271 chips, S-100 compatible, 350 nsec. access time, 550 nsec. cycle time, transparent refresh, no wait states for 2 MHz 8080 processor, on board clock, 5 watts power consumption, 1 MHz direct memory access, 16k addressing, solder masked, assembled with sockets, tested, burned in, guaranteed one year.

Dynabyte

Builders of the Great Memory



Circle 43 on inquiry card.

Part 2:

Simulation of Motion An Automobile Suspension

Stephen P Smith
POB 841
Parksley VA 23421

Have you ever taken your system out to a club meeting or demonstration, only to find that something is ruining your car's handling? Was it because of the heavy power supply in the back seat? Would heavy duty shock absorbers help? You can answer these questions using your personal computer and the simulation techniques found here.

Last month *[page 18]*, I introduced some basic ideas used in simulating motion. A games application was used as an example. This month I'll expand on that base, explain some additional ways that forces can act, and demonstrate a more accurate technique for computing speeds and positions. The example I'll use will be a simulation of an automobile suspension and its response to a varying road surface. Automobile enthusiasts will be able to see how different springs and shock absorbers would affect the way a car rides. More important, all computer users will acquire some additional tools to use in their own simulations and gain insights into new applications for their personal systems.

First, let's review the basic points made in the last article. When beginning a simulation, you will first divide the motion being simulated into degrees of freedom. In other words, you will decide which motions you want to simulate, up and down, side to side, etc. From then on, calculations will be made separately for each degree of freedom. Next you will decide which forces are acting in each direction and determine how much each force would change the speed of some object in 1 second. If you use the metric system of units, the change, or acceleration (in meters per second per second), will be exactly equal to the force (in newtons) divided by the mass of the object (in kilograms). You will now be ready to predict the speed and position of the object at a step of D seconds into the future. Add up the effects of the individual forces. Multiply the total by D (the step size) and add the product to the present speed. This is the speed of the object at a time D seconds into the future. Now multiply the speed by D and add that product to the present position. This is the position the object will take in D seconds. The simulation program will now calculate new values for the forces and mass and step the simulation forward once more. The process will continue until an end condition is reached.

In the lunar lander game simulation, two degrees of freedom were considered, up and down, and side to side. The up and down or vertical motion was affected by gravity and thrust. The side to side or horizontal motion was affected only by thrust. Both of these forces were determined independent of the speed and position of the lander. Gravity provided a constant change in speed, and thrust was controlled by the user. In this article we will explore variable forces which are not determined by the user, but directly by the speed and position we are simulating.

As mentioned earlier, the example we'll use is an automobile suspension, the parts which connect the wheel to the body. The most important of these parts are the spring and the shock absorber. We will assume that there are other parts which keep the wheel from moving back and forth, but only the wheel's up and down motion will be considered (see figure 1). Of course, the entire car can also move along the road. We will consider that as a second degree of freedom. Let's examine separately the forces that contribute to vertical and horizontal motion.

Motion down the road results when the car's motor, through the wheels, pushes the car forward. Air resistance and rolling friction try to slow it down. To simplify the simulation, we will assume that these forces balance each other exactly. This means that the speed along the road will not change. If the speed starts at some value other than zero, the horizontal position will change. As we will see later, the

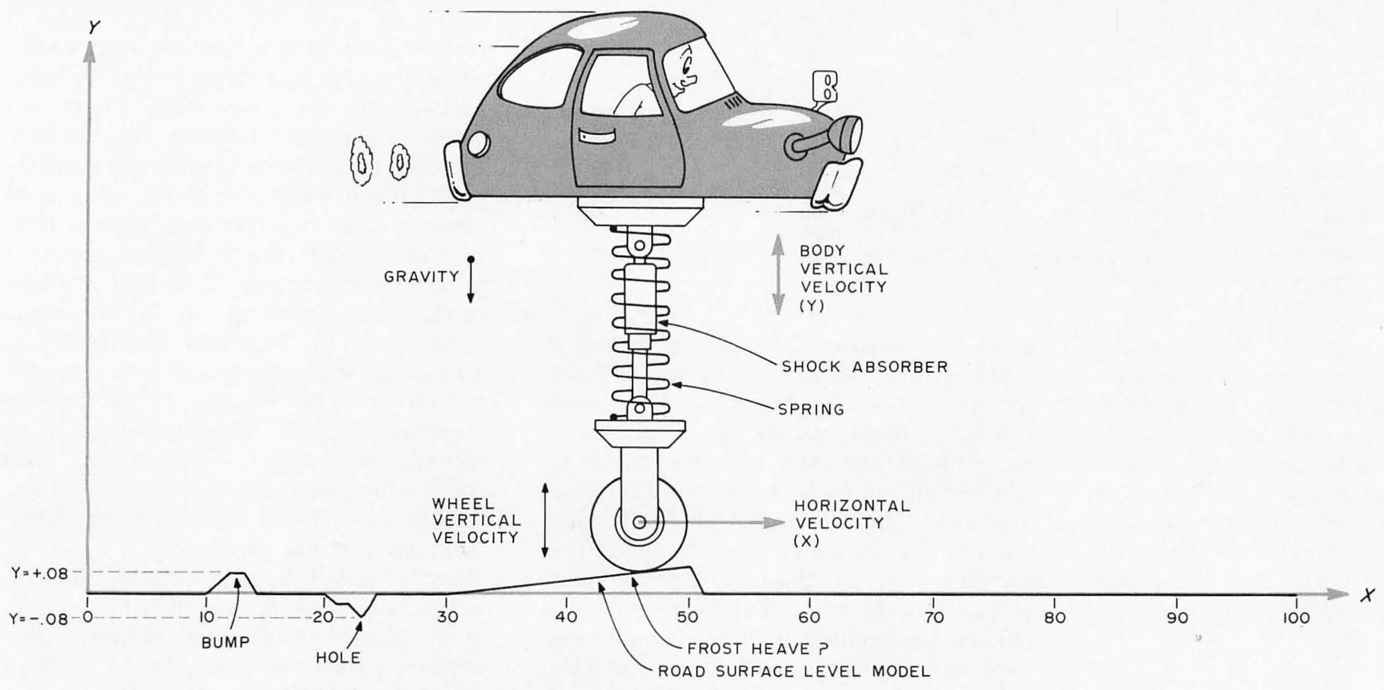


Figure 1: A conceptual model of the "automobile" (unicycle, rather) which is modelled in the sample program of listing 1 as discussed in this article. The wheel in this model tracks the road surface exactly, and has its own vertical velocity due to the horizontal velocity interacting with bumps in the road. The actions of the wheel in turn couple through the spring and shock absorber suspension to the "body" of the automobile. The purpose of this simple model is to calculate the vertical position of the car body at any given point down the road, given the effects of gravity, shock absorber, spring and excitement provided by the bumps and holes. The table in the figure is taken from lines 605 and 606 of the BASIC program of listing 1, and is used to plot the road surface. A better dynamic model of a car would have many more degrees of freedom than this simple model.

simulation program must keep track of the position along the road, because it will determine how the wheel, and in turn the body, moves up and down.

In the vertical degree of freedom we will need to consider gravity. You will remember from the last article that to simulate gravity a program subtracts a constant value from the speed for each unit step. (Speed and position are considered positive if they are directed upward.) On the earth the gravitational acceleration constant is 9.8 meters per second per second, so for each second of simulated time, velocity changes by 9.8 meters per second. Since the car obviously does not continue to move downward, there must be other forces balancing gravity. These are produced by the spring and shock absorber, and are determined by the vertical speed and position of the body.

Let's examine the spring first. At its normal length (often called the free length) a spring produces no force at all. If it is compressed, in other words forced to become shorter, it will push back on whatever is compressing it. The shorter the spring is forced to become, the harder it will push back. This is an example of a force that depends upon position. In the automobile example, as gravity pulls the body down, the spring is compressed. The spring begins to push upward on the body, and at some point the two forces balance each other. The body will eventually come to rest there.

Knowing a little information about the spring we can compute that point. Springs produce forces which are equal to the distance they are compressed times a constant. The metric units for the constant are newtons per meter. Sample values are shown in table 1, column a. Suppose that gravity exerts a force of 5000 newtons on the car body; then a spring with a constant of 100000 newtons per meter would have to be compressed .05 meters (100000.x.05=5000.) to balance the pull of gravity. At this point the system would be in equilibrium.

What about the shock absorber? It was designed to produce a force that depends not on how far it is compressed, but on how fast it is being compressed. The faster you try to move it, the harder it resists being moved. Like the spring, a constant is used to calculate the force, this time multiplying the speed. The metric units for this constant are newtons per meter per second and some representative values are shown in table 1, column b. At equilibrium there is no motion, so the shock absorber produces no force. If you were to push down the car

Vehicle	(a) Spring	(b) Damping
Full size car (LTD, etc)	3200	1450
Intermediate (Torino, Cutlass, etc)	3000	1200
Compact (Nova, Aspen, etc)	2800	1000
Subcompact (Vega, Pinto, etc)	2600	700
Add 20% for heavy duty suspension.		
Subtract 20% for front wheel.		

Table 1: Representative spring and damping constants for automobiles. The units are metric: the spring constant is quoted as newtons per meter of compression; the dampling constant is expressed as newtons per meter per second.

body at a speed of 2 meters per second, a shock with a constant of 50 would resist that motion with a force of 100 newtons (50 x 2). When you let up on the body, the spring would exert a greater force than gravity and the body would move upward. The shock absorber would also resist that motion. This action is called damping. The damping in an automobile suspension must be carefully chosen so that the body returns quickly to equilibrium, but does not continue to bounce back and forth for very long afterward.

Armed with your present knowledge of simulation you should be ready to make just such a choice using a trial and error approach. Calculate the forces on the body, and then use them to find the speed and position one step into the future. That speed and position will be used to calculate new values for the forces, which in turn will be used to step the simulation forward once more. Repeating the process continuously, you will simulate the motion of the car body. Try different values for the spring and damping constants until the desired output is achieved for a given set of inputs.

The inputs, you'll remember, are going to be determined by the simulated position in the horizontal degree of freedom. At each position along the road the input routine will determine the height of the road surface above or below normal. If we assume that the wheel does not leave the road this will also give us the up and down motion of the wheel. The data can be stored in a table in memory. By entering different values for the horizontal speed at the start of the simulation, we can also vary how fast the car will pass over our model road. At each step the program will enter the table to find the road height which corresponds to the current horizontal position.

This method will work as long as there is an entry in the table for every horizontal position we will find. That could be a very big table, especially if the step size is small. To eliminate the need for large tables, we can use a technique called interpolation. Very simply, interpolation is done like this. When the program enters the table, but doesn't find an entry exactly equal to the current horizontal, it finds the next smaller entry and the next larger entry. An interpolation formula is then used to figure out where the present position falls between the two table entries, and to calculate the road surface which lies at the same point between the corresponding table entries of road height. For example, suppose a program entered table 2 to find the road surface corresponding to a horizontal position of 11. It would find entries at 10 and 12 with corresponding road heights of 0.0 and 0.08. Because 11 lies halfway between 10 and 12, the interpolation formula will find a corresponding road surface that lies half way between 0.0 and 0.08 or 0.04. There are other interpolation formulas that use three, four, or more of the table points, but this method using two is generally accurate enough with a reasonably detailed table. To simplify your implementation of interpolation, I have included a BASIC function in the program of listing 1 which uses the 2 point method. Users can simply place their own tables in the data statement and use the function in their programs, or they can follow through the equations and implement them directly.

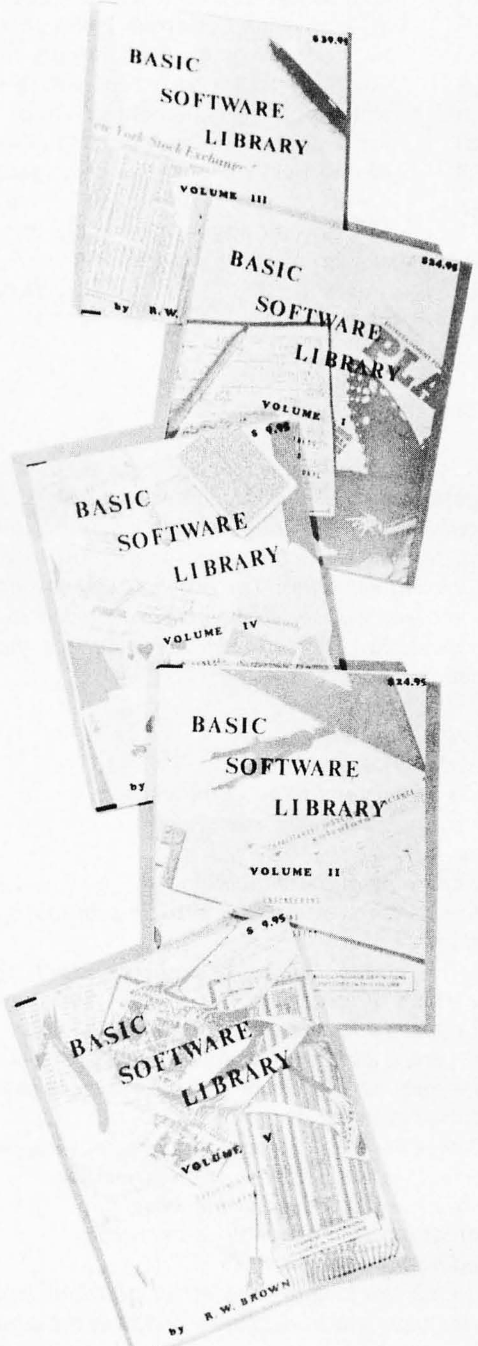
Table 2: A sample road surface table. This table is used to draw the surface curve shown in figure 1.

Horizontal Position	Road Surface
0	0.0
10	0.0
12	0.08
13	0.08
14	0.0
20	0.0
21	−.04
22	−.04
23	−.08
24	0.0
30	0.0
50	0.1
51	0.0
100	0.0

In our automotive simulation, the interpolated table data will give us the vertical road and wheel position. The difference between this and the vertical position of the body will be the amount the spring is compressed. We can quickly calculate the resulting force. If the simulation program retains the wheel's position from the previous step, it can also calculate the wheel's vertical speed. Reversing the equation used to find a new position, the speed is equal to the difference in the two positions divided by the step size. If the wheel moved from .08 meters to .04 meters in a step of 0.01 seconds, its speed would be $(.04-.08)/.01 = -4.0$ meters per second. The difference between this speed and the speed of the body is used to calculate the force produced by the shock absorber. All these calculations are included in the BASIC program of listing 1. Readers who want more detail on the equations will find them there as well-commented program statements.

Also in that program is a new method for computing speeds and positions. The equations used in the lunar landing game worked fairly well when the forces did not depend upon the speed and position. In this simulation they do, and even small errors can snowball if not corrected. To do this, we will use a powerful numerical technique, one which uses the results from three previous steps to help predict the next, and

Application Software!



You can buy software from anybody - but ours works in your system.

We only sell one product, Quality.

We have been in business for over nine years building a reputation for providing a quality product at nominal prices — NOT what the traffic will bear. Our software is:

- **Versatile** — as most programs allow for multiple modes of operation.
- **Tutorial** — as each program is self prompting and leads you through the program (most have very detailed instructions contained right in their source code).
- **Comprehensive** — as an example our PSD program not only computes Power Spectral Densities but also includes FFT's, Inverse-transforms, Windowing, Sliding Windows, simultaneous FFT's variable data sizes, etc. and as a last word our software is:
- **Readable** — as all of our programs are reproduced full size for ease in reading.
- **Virtually Machine Independent** — these programs are written in a subset of Dartmouth Basic but are not oriented for any one particular system. Just in case your Basic might not use one of our functions we have included an appendix in Volume V which gives conversion algorithms for 19 different Basic's; thats right, just look it up and make the substitution for your particular version. If you would like to convert your favorite program in to Fortran or APL or any other language, the appendix in Volume II will define the statements and their parameters as used in our programs.

Over 85% of our programs in the first five volumes will execute in most 8K Basic's with 16K of free user RAM. If you only have 4K Basic, because of its' lack of string functions only about 60% of our programs in Volumes I thru V would be useable, however they should execute in only 8K of user RAM.

All of our programs are available on machine readable media. For those that have specific needs, we can tailor any of our programs for you or we can write one to fit your specific needs.

Vol. I — $24.95 Bookkeeping Games Pictures	**Vol. III — $39.95** Advanced Business Billing, Inventory Investments Payroll
Vol. II — $24.95 Math/Engineering Plotting/Statistics Basic Statement Def.	**Vol. IV — $9.95** General Purpose **Vol. V — $9.95** Experimenter's Program

SCIENTIFIC RESEARCH

220 Knollwood, Key Biscayne, FL 33149

Phone orders call (800) 638-9194
Information—(305) 361-1153

Add $1.50 per volume handling, all domestic shipments sent U.P.S. except APO and P.O. Box which go parcel post. Foreign orders add $8.00/volume for air shipment and make payable in U.S. dollars only.

AVAILABLE AT MOST COMPUTER STORES

Master Charge and Bank Americard accepted.

Our Software is copyrighted and may not be reproduced or sold.

which then goes back and corrects the step when the predicted results are available. It is called, logically enough, a predictor-corrector method. Rather than attempt to explain it here, I'll provide a BASIC programming example which you can adapt to your own simulations. Readers with a good background in math may wish to reference a book on numerical methods for more details. In either case you will have acquired a tool which will be very useful in future simulations.

Looking back over the two articles you should begin to see some ideas for your own simulations. They could involve forces which are constant, user controlled, or which depend directly on the motion you are simulating. Inputs can come from your keyboard, from an analog device such as a joystick, or from tables interpolated by your program. The outputs might tell you how well you are playing a game, or which of several configurations is best for a design you are contemplating. In the next article I'll continue to expand on the types of forces considered. In particular, I'll show how you can handle forces which act in more than one degree of freedom and suggest some ways to handle rotary motion.■

Automobile Suspension Simulator

```
100 REM SET PROGRAM CONSTANTS
110 K1=-100000
120 K2=-2000
130 M=500
140 V=10
150 D=0.05
151 REM SET INITIAL SPEED AND POSITION
160 T=0
170 P1=9.8*M/K1
171 S1=0
172 REM FIND INITIAL ROAD SURFACE
175 I1=0
176 X1=0
177 Y1=0
178 X=0
182 GOSUB 600
183 I2=(Y-I1)/D
184 I1=Y
185 REM CALCULATE INITIAL FORCES
186 F1=(P1-I1)*K1/M
187 F2=(S1=I2)*K2/M
188 A1=F1+F2-9.8
189 REM SET PAST DATA EQUAL TO INITIAL DATA
190 S2=S1
200 S3=S1
210 S4=S1
230 A2=A1
240 A3=A1
250 A4=A1

259 REM     BEGIN SIMULATION
260 REM PREDICT SPEED AND POSITION
270 S=S1+D/24*(55*A1-59*A2+37*A3-9*A4)
280 P=P1+D/24*(55*S1-59*S2+37*S3-9*S4)
281 X=X+V*D
282 REM FIND NEW ROAD SURFACE
283 GOSUB 600
284 I2=(Y-I1)/D
285 I1=Y
290 REM PREDICT NEW FORCES
291 F1=(P-I1)*K1/M
292 F2=(S-I2)*K2/M
300 A=F1+F2-9.8
310 REM CORRECT SPEED AND POSITION
320 S=S1+D/24*(9*A+19*A1-5*A2+A3)
330 P=P1+D/24*(9*S+19*S1-5*S2+S3)
340 REM CORRECT FORCES AND UPDATE SAVED DATA
341 F1=(P-I1)*K1/M
342 F2=(P-I2)*K2/M
350 A4=A3
360 A3=A2
370 A2=A1
380 A1=F1+F2-9.8
390 S4=S3
400 S3=S2
410 S2=S1
420 S1=S
430 P1=P
440 T=T+D
450 PRINT T,S1,P1
460 IF X<100 THEN 270
470 END

600 REM INTERPOLATE TABLE TO FIND
601 REM  VALUE OF Y CORRESPONDING
602 REM  TO GIVEN VALUE OF X
605 DATA 0,0,10,0,12,0.08,13,0.08,14,0,20,0,21,-0.04
606 DATA 22,-0.04,23,-0.08,24,0,30,0,50,0.1,51,0,100,0
611 REM TABLE FORMAT IS X(1),Y(1),X(2),Y(2), ...
620 IF X<X1 THEN 670
630 X2=X1
640 Y2=Y1
650 READ X1,Y1
660 GO TO 620
670 Y=Y2+(Y1-Y2)*(X-X2)/(X1-X2)
680 RETURN
```

Listing 1: This program was written to help interested readers follow the mathematics of the accompanying article. Particular attention should be paid to the interpolation subroutine and to the equations for the predictor-corrector method of predicting future positions and velocities. The program was not intended to be efficient; readers will surely be able to shorten it once the method is understood. The following table defines the variable names I've used.

- *K1 = spring constant*
- *K2 = damping constant*
- *M = mass supported by the spring*
- *V = horizontal speed of the entire car*
- *D = time step size*
- *T = elapsed time in the simulation*
- *P, S, A = predicted values for vertical position, speed, and total effect of forces*
- *P1, S1, A1 = present values of vertical position, speed, and total effect of forces*
- *S2, A2 = speed and effect of forces one step past*
- *S3, A3 = speed and effect of forces two steps past*
- *S4, A4 = speed and effect of forces three steps past*
- *F1 = change in speed due to spring*
- *F2 = change in speed due to damping (shock absorber)*
- *I1 = current vertical position of the wheel*
- *I2 = current vertical speed of the wheel*
- *X = current position of car along the road*
- *Y = road height at position X*
- *X1, Y1, X2, Y2 = table entries for positions immediately greater than and immediately less than the current value of X*

I expect it will occur to many of you that graphic rather than printed output will make this program much clearer. The waveform produced by a plot of the data would give you a much better feel for the motion of the car body. For the example in this listing, try plotting position from −0.1 meter to +0.1 meter versus time from 0 to 100 seconds.

One final note: to avoid losing data, it is important that the interval between points of the table in the interpolation subroutine is larger than the distance the car moves in one step. In other words, if you want to model a road that changes rapidly, you will have to reduce the step size (D) to a value less than the minimum of (X(n) − X(n+1))/V.

Ximedia Presents . . .

a reliable small business disk system for both word processing and computation.

Ximedia Systems stress reliability and performance at an affordable price. That requires hundreds of product evaluations. It also requires a strong commitment to selling only those components which have operated dependably in business or professional applications. The Small Business Disk System (SBD-I) is that kind of system.

OPERATIONAL CAPABILITY

The SBD-I system is designed for the professional for such tasks as word processing, accounting, inventory control, and other business-related activities.

We like this system for a number of reasons. First, the North Star Disk-Operating-System (DOS) with extended disk BASIC is excellent. It's as bug free as any BASIC now in use. Also, it's a truly extended BASIC with features like formatted output, callable machine language subroutines, random and sequential file access, and a powerful line editor.

On the hardware side, we have found all the North Star products to be well-engineered, well-designed, and extremely reliable. The Micro-Disk System, around which the HORIZON is built, was the first mini-floppy system commercially available, and is still the leader. Each 5½" diskette will hold 90K bytes of data, which means about 50 pages of text per disk, perfect for word processing. We've included 32K of Crea/Com dynamic memory which is in our experience, one of the better dynamic boards around. They require low power, they're fast, and they use a good refresh scheme. For memory, 32K should be plenty and the HORIZON is expandable up to 65K and beyond!

The combination of 32K of Ram, 180K of Disk, Z-80 speed and the FOX-1100 make SBD-I system a powerful tool for the professional user. It's a good medium-sized system that can be expanded to include more memory, more disks, and other S-100 accessories when and if they become necessary.

TECHNICAL CAPABILITY

The North Star HORIZON computer is built around the highly successful Micro-Disk System. The CPU card is based on the Z-80 microprocessor, and designed to run at a full 4 MHz. It also features a jump-on-start circuit to pull in the disk bootstrap on power-up.

The HORIZON motherboard has an on-board serial interface with room to add another serial and/or parallel I/O port. The Crea/Com memory is a very fast (250s) dynamic ram board with lots of features. First, the refresh is handled by an on-board processor and is *totally* invisible to the main processor. Other goodies include a pin 67 phantom line, full or partial write protect, no wait states (even at Z-80 speeds), and guaranteed tested, burned in factory prime chips.

The Perkin-Elmer FOX-1100 is a super high quality, low cost CRT terminal that's a cut above the competition. The first big improvement is the 9x12 matrix. This means terrific readability *and* lower case characters that descend! Other features include a hooded, eye-level display, black-on-white or white-on-black display, full 96 character ASCII character set, direct cursor addressing, 80x24 display, resettable tab stops, Typamatic auto repeat, wraparound or scroll modes, and a debug mode for displaying control characters. All in all, the FOX is a superior terminal designed for the serious user and built to take abuses of day in and day out use.

The alternate SBD-IP system includes a 55 CPS daisy-wheel printer for letter quality hard-copy output.



SMALL BUSINESS DISK SYSTEM SBD-I

QUANTITY	DESCRIPTION	SUGGESTED RETAIL
1	North Star HORIZON—2 computer (without memory board)	$1,949.00
3	Extra card guides and connectors	21.00
1	Crea/Com 32K fast dynamic memory	885.00
1	Perkin-Elmer FOX-1100 CRT terminal	1,440.00
1 set	Interconnection cables	25.00
1 set	Ximedia proprietary system documentation	50.00
	TOTAL SUGGESTED RETAIL	$4,370.00

Fully Assembled and Tested XIMEDIA SYSTEM PRICE—$3,975.00

CONVENIENT ORDERING: Toll free number. Prepaid, Mastercharge, and Visa orders shipped free; others freight-collect. Most orders shipped from stock. If not, we will notify. California residents add 6% sales tax.

XIMEDIA
1290 24th Avenue
San Francisco, California 94122
(415) 566-7472 • Toll Free (800) 227-4440

Circle 132 on inquiry card.

A Little Bit on Interrupts

Robert R Wier
POB 9209
College Station TX 77840

While talking with fellow enthusiasts attending meetings of computer clubs, there seem to be several aspects of small computer systems which are particularly confusing to newcomers to the hobby. One of these is interrupts. This article explains how the mechanisms of interrupts work, and what can be done with them in a personal computer system.

History

When computers first came into widespread use, they ran primarily on card or tape batch principles. The operator had long lists of instructions which told him which card decks to use to run which jobs. Each job had to be set up independently, which was okay as long as this setup time was short in relation to the amount of time each job ran. A desired goal was to keep the machine running as much as possible. As technology advanced and job run times became shorter, setup time became a significant fraction of the total job run time. It was clear that if the machine could take over some of the chores of the operator, but at machine speed, the utilization of the system could be increased. Accounting and setup procedures could be accomplished by programs stored inside the machine, and then the computer could request the operator to perform only those duties that actually required human intervention (such as mounting a disk pack). Thus programs called "operating systems" came into use. About this same time, it was realized that if such a machine were going to run jobs under an operating system, there had to be some way to return control to the operating system should the program encounter difficulties. That is, the operator should be able to jerk control of the machine away from the program currently running and give it to the operating system without having to go through the process of clearing the machine and reloading the operating system manually each time. Another problem emerged at this time with the fact that as the central processing unit improved in efficiency due to the faster technology, the devices used for input and output, called peripherals, remained at about the same speed. Therefore, if the central processing unit had to wait for the completion of an input or output operation, it would just sit there testing and retesting to see if the program could proceed. This was frequently called a "busy wait loop," or "spin lock." It is a technique which is still frequently used in microprocessor systems.

Clearly, since IO operations were so slow, it would be nice if the processor could simply request the IO hardware to input to or output from memory directly without processor intervention. Then the processor could go on and perform useful computations while the IO operation was in progress. Of course this required considerably more sophisticated IO hardware than was in use previously, when the processor orchestrated every data transfer. But since the IO hardware didn't need to be able to perform complicated arithmetic functions it could be regarded as a "mini" central processing unit or (Aha!) microprocessor. Indeed, the original purpose of the microprocessor chip which has made our hobby possible was to produce cheap, reasonably smart peripheral systems at low cost. That is, each IO channel would have its own smaller processor to handle only data transfers between an IO device and memory. A little thought will reveal a problem, however. If the processor simply starts an IO operation and then pursues other matters, how does it know

when the IO operation is finished so that it may use the data input, or refill the buffer just output. What if the drive mangles the tape and the data has to be output or input again? What was needed was the ability for the IO processor to be able to tap the central processor on the shoulder and say "I'm finished," or "I fouled it up."

There was also the problem of real time applications which, depending on the system, needed the computer to be able to detect some condition, make a decision, and act on it quickly. If you had a big busy wait loop, where several instructions were executed in the loop between each checking of the status of each separate input signal, your refinery's catalytic converter might go critical before the computer even checked to see if something was wrong, a disquieting development.

What Interrupts Do

So, interrupts were devised. Indeed, some computer scientists feel that the major difference between the second and third generation machine was not only the transition to integrated circuitry, but the advent of the interruptable machine as well. But exactly what happens when something from the outside world, or a condition internal to the processing, wants attention?

Suppose the processor is hardwired with at least one interrupt line, and probably more. When an interrupt occurs, the desired effect is to:

- Store *all* the information regarding the presently running program which is necessary to resume execution at the same point some time in the future, so as not to have to start it over again from the beginning. This includes the program counter, any status information, and, optionally, the processor registers. This "state saving" activity must be complete or unpredictable behavior can ensue upon return to the interrupted process.
- Insert into the program counter the address of the first instruction in the interrupt program which will handle the condition which caused the interrupt. When the interrupt routine is finished, the status register(s), program counter, and processor registers of the interrupted program may be restored and the interrupted program resumes

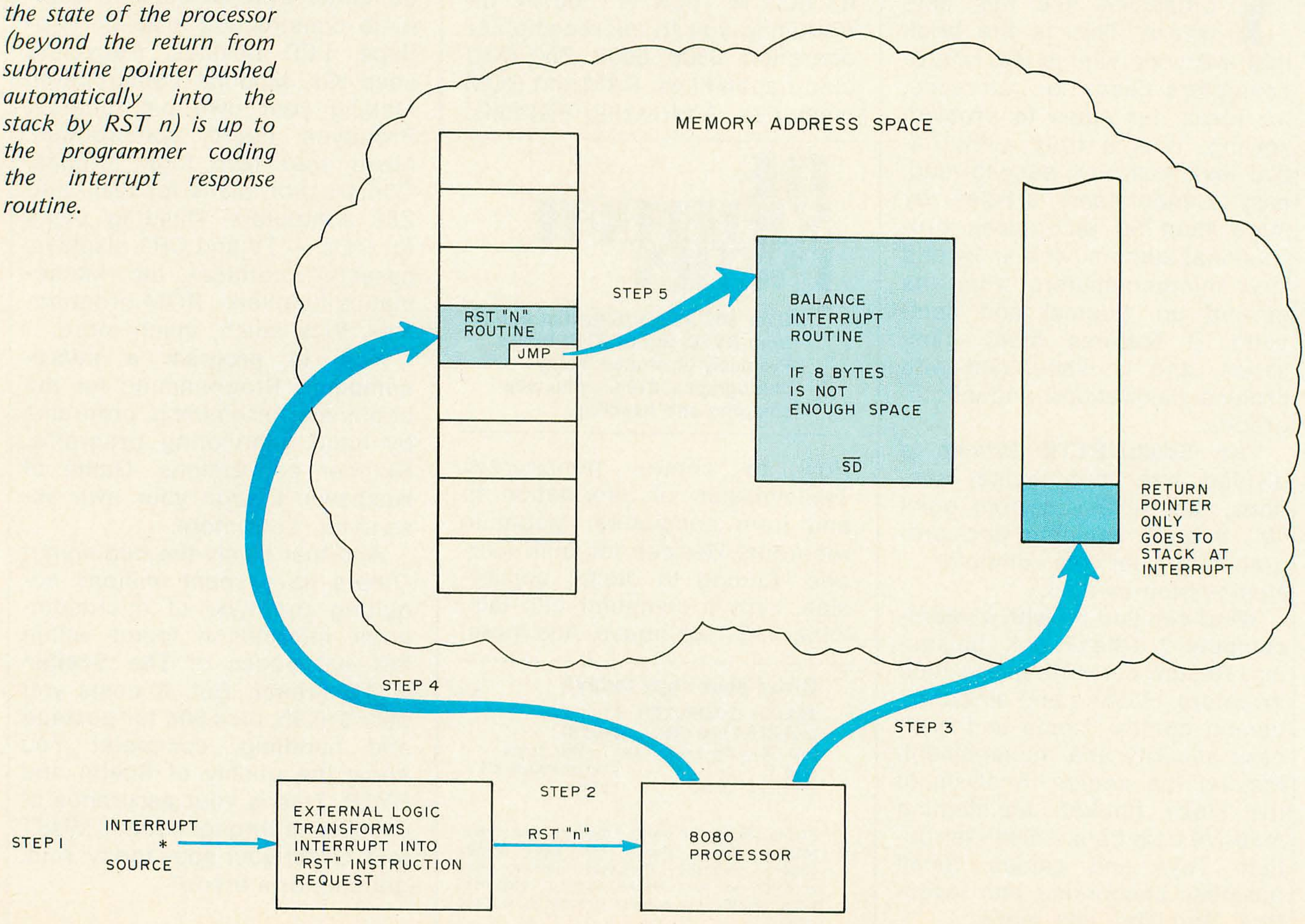


Figure 1: What happens when an 8080 interrupt occurs. The interrupt signal occurs first at some external device. Then, external circuitry creates an interrupt signal and sends an RST instruction to the processor as the second step. As a third step the old program counter information is saved on the stack to allow later return. Then, the fourth step which is part of executing the RST instruction is to jump to one of eight possible restart locations in the first 64 bytes of memory address space; if the eight bytes are not sufficient, step five, shown here, is a jump to an extension of the interrupt routine. Responsibility for saving the state of the processor (beyond the return from subroutine pointer pushed automatically into the stack by RST n) is up to the programmer coding the interrupt response routine.

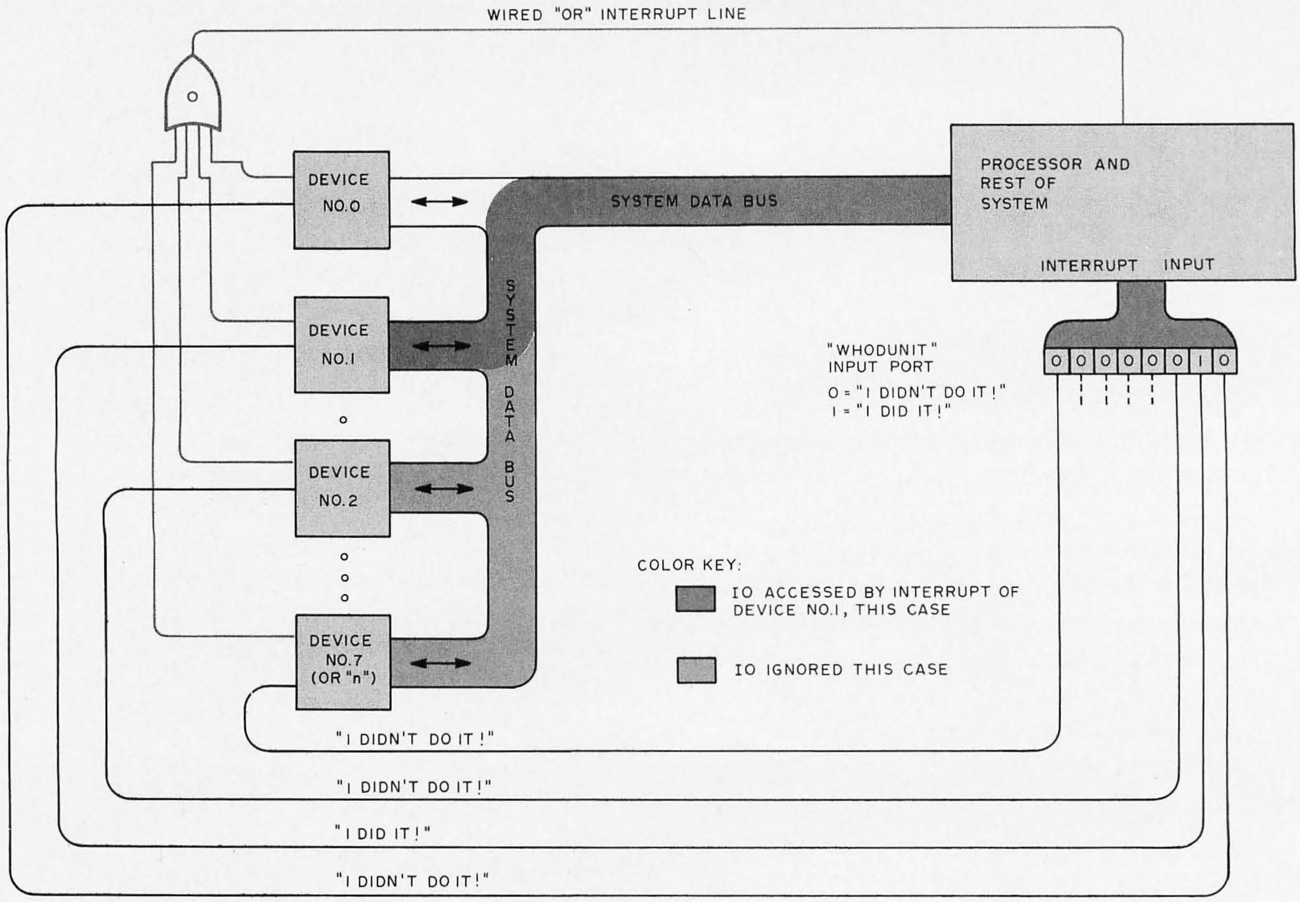


Figure 2: The "Who Done It" problem on interrupts. Some means must be provided to determine which IO device requested service when more than one device shares an interrupt line on any processor. Here is one way of determining "who done it": The input port "WHODUNIT" looks at eight single bit status flags corresponding to up to eight devices; if the flag is on, then that device "did it."

running without really being aware that it was temporarily not in control of the processor. This process of restoring the machine state is the inverse of the state saving activity.

Interrupt Hardware

The actual hardware that is included to effect interrupts varies somewhat from one processor to the next. Virtually all of them save the old program counter in some specified location and insert the address of the interrupt handler's first instruction into the program counter. This is in effect an unconditional branch to a subroutine with linkage for return after interrupt processing. Each machine is different, though, in the actions taken beyond these two basic functions. In the IBM 370 series, the hardware does practically everything for the programmer. In microprocessors, the software interrupt program must do some of the things that the hardware does in the larger machines. Let's look at the most popular microprocessors and see what they do.

Interrupts on the Intel 8080

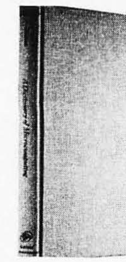
When an interrupt request is received, the 8080 completes the current instruction before taking any action on the interrupt. Virtually all miniprocessors and microprocessors do this, since there would be all sorts of problems encountered if an interrupt were recognized in the middle of the execution of an instruction. A little thought will show why. The 8080 does not increment the program counter. The program counter for the old program is pushed, saved, onto the stack. The next instruction to be executed is "jammed" onto the data bus by external interrupt circuitry and is called the restart instruction. Depending on the restart instruction operand, the next instruction executed (ie: the address placed into the program counter) may be one of eight possible decimal memory locations: 0, 8, 16, 24, 32, 40, 48, or 56. 8080 programmers will note that there are just enough memory locations, eight, between these addresses to save the registers of the old program, disable further interrupts, and execute a jump to another location, which in this case will be the interrupt service routine. This entire operation is explained in figure 1.

Obviously, if you ever contemplate using all eight classes of interrupts, you want to be sure not to program using the first 64 memory locations since those are reserved by the hardware for interrupt handling. But what if you want to have only one class of interrupt? Say, a panel switch that the operator (you) can push to get the attention

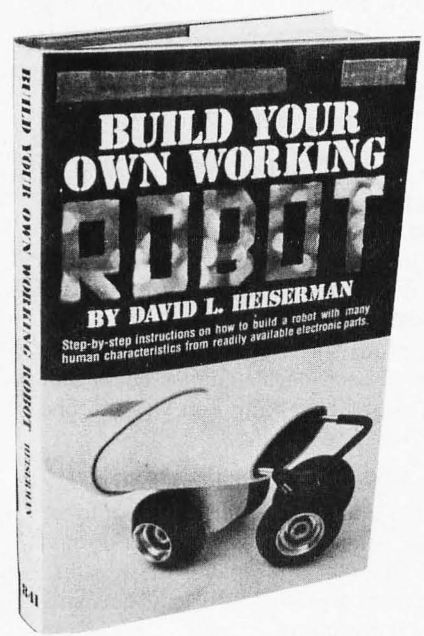
References and Robots from BITS

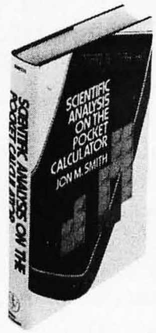


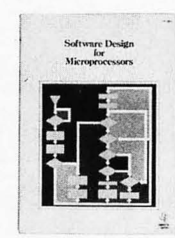
___Game Playing With Computers by Donald D Spencer, published by Hayden. What does it mean to play games using a computer? Read this book to get an introduction into numerous recreational uses of the computer to program and play mathematical and logical games. Topics include numerous mathematical problems, casino games, board games, unusual gambling games, and miscellaneous logic games. Numerous BASIC language programs and listings are included to show details. $16.95.



___A Dictionary of Microcomputing by Philip E Burton. In the opinion of BYTE's editor, Carl Helmers, "This is one of the best *designed* and executed dictionaries of computer related terms yet seen on the market. It is of particular relevance to those individuals who want a good general reference to numerous technical terms, broadly covering hardware and software fields as currently practiced." This new hardbound edition is part of the Garland Reference Library of Science and Technology. $12.50.







___Scientific Analysis on the Pocket Calculator by Jon M Smith, published by John Wiley & Sons. This book is another in a set of source books for mathematical analysis using the contemporary products of technology. It is oriented to the pocket calculator, yet it will provide you with algorithms and methods useful with any personal computer which implements the scientific and analytical functions found on a good pocket calculator. For a more complete description, see the book review on page 120 of the December 1976 BYTE; or order its 392 pages of detailed technical information and review its use for yourself. $13.75.

___Build Your Own Working Robot by David L Heiserman, published by Tab Books. This book will not tell you how to build Robbie, the robot of **Forbidden Planet**, or a classical android of science fiction. What it will introduce you to is the problems of making a robot mobile device called Buster III, using pre-microprocessor TTL integrated circuits for all logic functions. It is a must book for background reading, but much of the logic can be extremely simplified using today's microprocessor technology. Use this book as a first look at these problems from which you can build further and more elaborate solutions. Softbound, $5.95.

___Software Design for Microprocessors. This stand alone guide to microprocessors has been designed by the people at Texas Instruments to convey knowledge to the first time user of microprocessors. This excellent source book of computer concepts begins with an outline of the basic principles of the general purpose computer, its machine architecture, software, and methods of addressing. It proceeds to discuss how to build software, what is involved in documenting what you've done once you've done it, the mechanics of programming, and specific examples using the TI TMS-1000, TMS-8080, TMS-9900 and SBP0400 designs. You'll find a thick hardcover textbook filled with over 370 pages of useful information including a comprehensive glossary of microprocessor terminology, among several other detailed appendices. $12.95.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

Check payment method:
___My check is enclosed
___Bill my MC No. ______________ Exp. date ______
___Bill my BAC No. ______________ Exp. date ______

Total for all books checked $______

Postage, 50 cents per book for ______ books $______

Overseas, 75 cents per book for ______ books $______

Grand Total $______

Name ______________

Address ______________

City ______________ State ______ Zip ______

Signature ______________

Prices shown subject to change.

You may photocopy this page if you wish to leave your BYTE intact.

All orders prepaid.

In unusual cases, processing may exceed 30 days.

of the machine. Then just program the particular location that you (or the computer hardware designer) hardwired in. Let's suppose for a minute that you need more than eight interrupts. That's possible, within a few restrictions as shown in figure 2. Just OR the interrupt request lines from the outside world together and feed them to the same interrupt line going into the processor. But then how do you know which device has caused the interrupt? Obviously there will have to be another signal somewhere to indicate which device needs attention. This could be implemented in a variety of ways:

- The device could place an identifying number on the data bus which would identify the device.
- An input port could be wired so that the device would signal that it needed attention.
- The processor could send an interrogation to each device connected to that interrupt line asking if it was the one that sent the request.

The first and second methods are faster since the device number or input data could be used as an index to go to the appropriate interrupt handler program. The third method is called polling and may be somewhat time consuming if many devices use the same interrupt line. Because so much of the interrupt logic of the 8080 is external to the chip, there can be considerable variation. Most 8080 systems use a simple restart (RST) operation code, but any instruction including jump (JMP) or call (CALL) can be used with appropriate external logic.

Motorola 6800 Interrupts

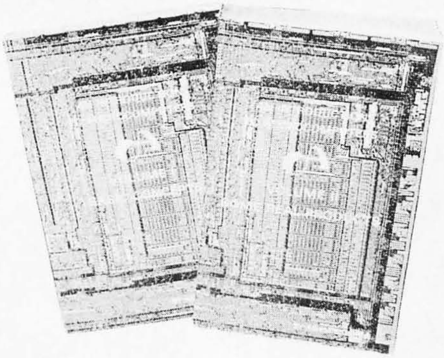
This chip has built into it the capability of decoding and servicing a smaller number of interrupts, but in a more automatic way than the 8080. The 6800 uses an indirect "vectored" interrupt situation in which each source of an interrupt looks up a unique vector location for the address of its service routine. When an interrupt is indicated to the 6800 by one of three possible sources, the processor automatically saves the two accumulators, index register, status register, and program counter on the stack, and in the process of doing so it changes the stack pointer. Thus, the 6800 has the advantage of never requiring program code to achieve state saving functions. It simultaneously has the disadvantage of always performing a complete state save so there is no way to "cut corners" and save time by ignoring the saving and restoring of data which is not changed by the interrupt routine. This vectoring method also has the disadvantage of requiring that the stack pointer must never be used for other purposes (such as a pseudo-index register) when interrupts are possible. The three interrupts possible on the 6800 are:

Maskable Interrupt (IRQ). This interrupt occurs when a hardware signal causes a low state on the IRQ line of the processor. This line is always wired in a "wired or" configuration when multiple sources are used, so some form of polling or priority logic is needed to identify sources. When an interrupt occurs, a flag is set in the processor which prevents a second interrupt from interrupting the routine which processes the first to arrive.

Non-Maskable Interrupt (NMI). This interrupt is identical to the IRQ interrupt except that no "masking" of repeated interrupts occurs in the processor to prevent conflicts. As a result, without external logic to do the masking only one interrupt source should be dedicated to this signal. Motorola intended this line to be used with the absolute highest priority external signal in a typical system: the signal which indicates a 110 VAC main power supply "power failure" in a dedicated application system; the interrupt response routine in such a case would have enough time before the capacitors of the power supply discharge (typically) to save the state of the processor and prepare for later return of power. But the intended use does not mean the only use, and with proper care this interrupt line can be used for inputs as diverse as a direct memory address (DMA) controller or real time clock.

Software Interrupt (SWI). This interrupt occurs when a program executes a software interrupt instruction. The actions taken are exactly the same as those for the totally asynchronous NMI and IRQ hardware inputs. The only difference is that the SWI is not a true interrupt since it is programmed into the software at a fixed point, whereas an interrupt such as NMI or IRQ can occur at any time relative to the execution of a program. The SWI instruction, in effect, is a call to an interrupt subroutine, with return implemented via an RTI (return from interrupt) instruction.

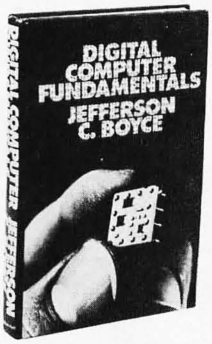
There is one further method of interrupting a process in the 6800 which is not characterized by the state saving needed to effect a true interrupt style action. This is use of the "reset" (RES) line of the hardware. This form of interruption merely causes an



Adam Osborne's books **An Introduction to Microcomputers, Volumes 1 and 2,** are a concise compendium of the technical details of microprocessors at the component (engineering) level. These are the source books for the system designer who plans to employ the microprocessor, or the advanced homebrewer who wants a dash of customization not found in commercial products.

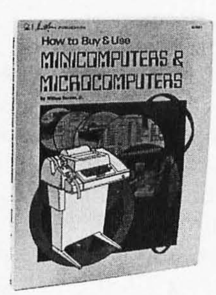
____ Volume 1 is subtitled "Basic Concepts." This is the book which presents a framework of ideas concerning the design and use of small computers implemented with LSI. Topics include definitions of the microcomputer, fundamental concepts of logic and numbering characteristics of instruction sets, etc. $7.50.

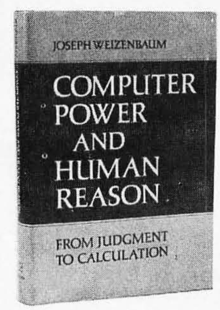
____ Volume 2 is a much thicker detailed volume which complements the information in the first volume. This is the volume which fills in many of the details left out of the conceptual treatment in Volume 1. Here you'll find 19 detailed chapters on the engineering and logical specifications of products made by 16 different manufacturers, including in many cases reprints from the manufacturers' documentation as well as new materials provided by the author. Published in 1976, it even includes such processors as the MicroNOVA by Data General and the Texas Instruments TMS-9900 as well as the older 8 and 16 bit machines. Organization is by design type, and where parts of several manufacturers were intended for a given processor design such as the 8080, these are grouped into a single chapter. Revised second edition $15.00.



____ **Digital Computer Fundamentals** by Jefferson C Boyce. The way to a world of learning is through books. A great place to start, and to return from time to time, is the classical textbook. This new book from Prentice Hall is intended as just that. Topics covered include digital computer operation, basic computer circuits and concepts, Boolean algebra, implementing computer operations in hardware, communicating with the computer and related issues of coding schemes, detailed discussions of the control section, memory functions, arithmetic and logic functions, input and output functions of a classical computer, a chapter on computer programming and a final summary chapter on the details of a typical minicomputer design interpreted in the light of the more theoretical general concepts in the book. This book is excellent background information for the literate and well read hacker. Order yours today. $15.95 hardbound.

____ **How to Buy and Use Minicomputers and Microcomputers** by William Barden. People have often asked us where to turn to get an introductory book about computers for personal use. One excellent place to start is **How to Buy and Use Minicomputers and Microcomputers,** William Barden Jr's instant summary of the small computer revolution, published by Howard Sams in mid-1976. This is one of the first books of the "general introduction to computers" genre to be published with an emphasis towards the small computer and personal computing as it is being practiced these days. the book, written for the novice as well as the expert, surveys the technical details of the field in nine chapters and 10 appendices. This book is light (but essential) reading for the experienced computer person, and worthy of serious, concentrated perusal by the novice. $9.95.





____ **Computer Power And Human Reason** by Joseph Weizenbaum. This book is one which should be purchased or read for several reasons. If you're presently a programmer by trade or skill, you'll see a philosophy of computer use and abuse propounded. It's genuinely interesting, and definitely provocative if you reference the storm of letters, counter letters and counter counter letters which this book produced in the Association for Computing Machinery's *SIGART* newsletters during 1976. If you're a novice to the field, the tutorial and explanatory chapters of this book, which are aimed at the layman, will serve as an excellent background source which is also eminently readable. This includes an excellent and low level explanation of what an algorithm is, and how computers go about executing effective algorithms. $5.95 softbound.

Send to:

BITS, Inc
70 Main St
Peterborough NH 03458

Total for all books checked $ ____

Postage, 50 cents per book for ____ books $ ____

Overseas, 75 cents per book for ____ books $ ____

Grand Total $ ____

Check payment method:

___ My check is enclosed

___ Bill my MC No. ________ Exp. date ____

___ Bill my BAC No. ________ Exp. date ____

Name ________

Address ________

City ____ State ____ Zip ____

Signature ________

You may photocopy this page if you wish to leave your BYTE intact. In unusual cases, processing may exceed 30 days

Circle 12 on inquiry card.

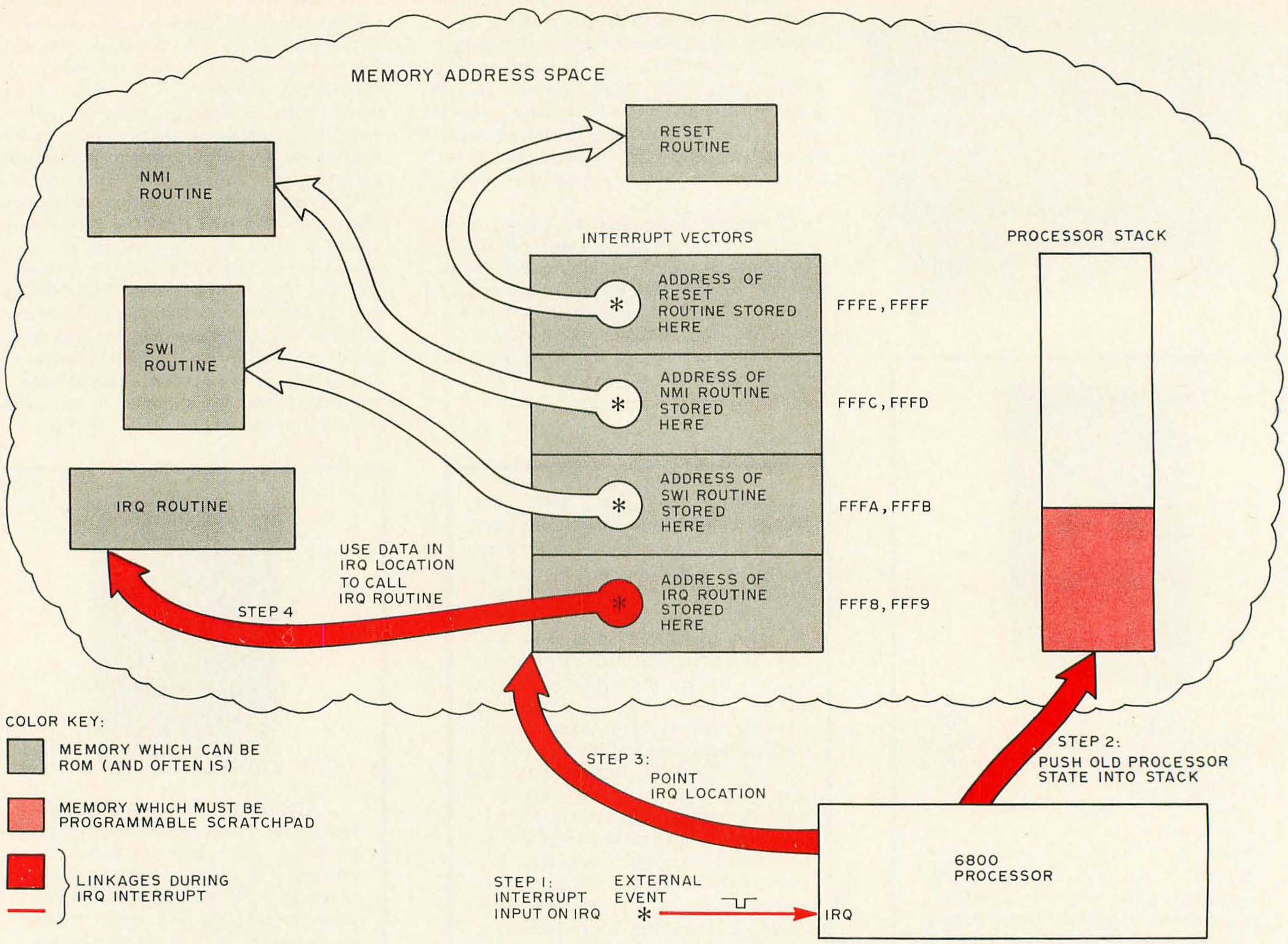


Figure 3: The 6800 processor's interrupt structure. This vectored interrupt method starts with an interrupt signal to the processor. In this example, IRQ occurs, so the processor generates a reference to the IRQ vector location at hexadecimal FFF8 and FFF9. The two byte address at the IRQ vector location in turn points to the IRQ routine somewhere else in address space and as the last step in the process, the routine is called. As part of the special interrupt routine call, the old state information is pushed onto the stack.

unconditional branch to a restart location, and is typically used to initialize the system or to recover from disastrous errors.

All four sources of interruption of the 6800 processor, IRQ, NMI, SWI and RES use a similar indirect vectored approach to locating the address of the desired routine. In the cases of IRQ, NMI and SWI the desired routine is a subroutine which returns via an RTI (return from interrupt) instruction; in the case of RES the desired routine is the beginning of the software which gains control when the processor is restarted.

In each case, the processor uses a 2 byte address stored in the region from hexadecimal address FFF8 to FFFF in memory address space as the starting address for the desired routine. Thus, for example, suppose a source of an interrupt changes the state of the IRQ line, causing an IRQ interrupt. The processor first completes the previous instruction, as noted earlier. Then, instead of executing the next instruction, it executes the details of the built in "state saving" sequence. After state saving, the processor sends out address to memory for location FFF8 from which it obtains the high order address of the interrupt routine. Then it sends out the address FFF9, from which it obtains the low order address of the interrupt routine. It then branches to the interrupt routine at the address just obtained. A similar process occurs for the NMI response using the data contained in locations FFFC and FFFD as an address; for the SWI response using data contained in locations FFFA and FFFB as an address; and for the RES response using data contained in locations FFFE and FFFF as an address.

The MOS Technology 6502

This 8 bit processor is very similar to the 6800 in its processing of interrupts. There is no separate vector for a software interrupt as implemented in the 6800, so the 6502's interrupt vector region only includes nonmaskable interrupts (FFFA and FFFB contain the address), reset (FFFC and FFFD contain the address), and maskable interrupts (FFFE and FFFF contain the

NOTICE

to all BYTE magazine subscribers and would-be subscribers

SAVE MONEY! As of January 1, 1978, we are raising our subscription rates to meet the increased costs of producing and mailing *BYTE* to our more than 110,000 monthly readers. The new domestic subscription rates, effective January 1, 1978:

Current Rates		Rates Effective January 1, 1978	
One year U.S.	— $12	One year U.S.	— $15
Two years U.S.	— $22	Two years U.S.	— $27
Three years U.S.	— $32	Three years U.S.	— $39

If you already subscribe to *BYTE* and still have several months to go before expiration of your subscription, you can still take advantage of current rates for renewal (even up to three years). Use coupon below. When your present subscription expires, your renewal order commences.

If you are not yet a subscriber to *BYTE,* the leading magazine for the creative home computer experimenter, don't delay . . . mail this coupon and start getting your own copies of this invaluable magazine, before new rates become effective.



USE THIS MONEY-SAVING COUPON TODAY

To new subscribers:

Read your first copy of *BYTE,* if it's everything you expected, honor our invoice. If it isn't, just write "CANCEL" across the invoice and mail it back. You won't be billed, and the first issue is yours.

Allow 6 to 8 weeks for processing.

BYTE Subscription Dept. • P.O. Box 361 • Arlington, Mass. 02174

PLEASE ENTER MY SUBSCRIPTION FOR:

☐ One year $12 (12 issues) ☐ Two years $22 ☐ Three years $32

☐ Check enclosed (entitles you to bonus of one extra issue)

☐ Bill me ☐ Bill BankAmericard/Visa ☐ Bill Master Charge

Card Number: ☐☐☐☐☐☐☐☐☐☐☐☐☐☐ Expiration Date: ______

Signature: ______ Name *(please print)* ______

Address: ______

City: ______ State/Country: ______ Code: ______

Offer expires December 31, 1977

FOREIGN RATES FOR ONE YEAR: *(Please remit in U.S. Funds)*

☐ Canada or Mexico $17.50 (Air *delivered)* ☐ Europe $25 (Air delivered)

☐ All other countries except above: $25 *(Surface delivery)*

Air delivery available on request

address.) The 6502's BRK instruction is similar to the 6800's SWI, except it uses the same vector location as the maskable interrupt (IRQ) rather than a separate address vector.

Interesting Uses

Now knowing about interrupts, what are their uses on the personal computer system, and what kinds of programming should we use with them? Probably a majority of users will not need to use interrupts at all, at least until they have several years programming experience. If you have an 8080, just be careful to write your programs around the critical interrupt locations in low memory addresses, in case sometime in the future you decide to start using them. If you have a 6800 and use a dedicated monitor such as JBUG or MIKBUG, much of your freedom to use interrupts is replaced by hardwired response vectors in ROM found at FFF8 to FFFF. Almost certainly if you plan on writing or using some type of operating system, the interrupt facilities will need to be used in the interrupt routines.

The use of interrupts for IO operations probably will not be a major application except in cases of direct memory access or fast peripherals. Personal systems tend to be strongly oriented to a memory conservative type of programming, since the cost of the processor hardware is so low to begin with, and the slowness of IO is not really a significant factor.

Real time applications are likely to abound in small systems. The timers that are included in some systems often operate by allowing the program to load a desired number, which is then counted down (or added up, depending on the hardware) independent of the processor. When zero is reached, the timer can generate an interrupt. This could be useful in such applications as keeping track of how long programs use the processor, allowing a player a limited amount of time to make a move in games like Star Trek, generating time of day applications and so on. A very interesting real time application of interrupts is in the use of light pens on oscilloscope graphics displays. This is one use of computers that many hobbyists, upon seeing it operate for the first time, feel is just this side of magic. Actually, when you consider how the oscilloscope display is generated, the mechanism is very straightforward. You may deduce that the computer, or IO device, must know where the beam of light is currently positioned on the scope's screen, or else it would be just a jumbled mess. Therefore, if a photosensitive device is placed close to the screen, when the light beam strikes the cell an interrupt may be generated. This interrupt may then cause the location of the beam to be noted by storing the current values in the counters used to control the beam.

Another extremely interesting application is the emulation of hardwired instructions. If the processor allows software or illegal instruction interrupts then software routines may be programmed which will produce the same effect as if the desired instruction had actually been included in the silicon on the chip. For example, suppose that you frequently needed an instruction which would, for some unfathomable reason, add the contents of all the registers and output them to a teleprinter. You could set up a subroutine in each program that required this action. But if you found that you needed this instruction frequently in every program you ran on the machine, another way of implementing this routine would be to place into the program code (that is, the program being run) something to cause an interrupt.

This interrupt would cause the interrupt routine to determine which action was desired, execute it, and then resume the interrupted program. Of course, the instruction would be executed much more slowly than if hardwired. Once the routine was finalized, it could be burned into read only memory, and from then on it would always be available for the programmer's use. The actual bit pattern inserted into the program, to cause the interrupt, varies with the processor. If there are unimplemented operation codes then you may simply choose one and use it to signify the new operation from then on. If unimplemented operation codes do not exist, or they cause the machine to "hang up" and not interrupt, then a software interrupt, called a "supervisor call" on the 370, may be used. This is somewhat less pleasing, however, since the code on the program listing will always look the same (ie: a software interrupt) and make debugging a bit more difficult. The 6800's SWI instruction with its separate vector is ideally suited to this use. Obviously, a byte would have to be stored somewhere which would signify to the interrupt routine which operation was desired. In a 6800 this would be accomplished by following the SWI instruction with the appropriate 1 byte code, and modifying the stack so that RTI returns control one byte past its normal point of return.

It is possible to reproduce a particular machine's entire instruction set on another

entirely different machine in this manner. This is frequently called "emulation," although the term is also used to describe this process being accomplished by microcode which, confusingly enough, is only remotely related to microprocessors.

Conclusion

We have seen that the use of interrupts allows computers to become more versatile than when they are dedicated to one program. Interrupts allow the machine to interact with the outside world while at the same time allowing it to pursue "its own" interests. Interrupts are useful for accomplishing things in ways which, while more difficult to program initially perhaps, may be worthwhile in the ease of application.■

REFERENCES

1. *Intel 8080 Microcomputer Systems User's Manual*, Intel Corporation, Santa Clara CA, July 1975.
2. *M6800 Systems Reference and Data Sheets*, Motorola Semiconductor Products Inc, Phoenix AZ.
3. *MCS6500 Microcomputer Family Programming Manual*, MOS Technology, Norristown, PA.

Using the PolyMorphics Video Interface

Wayne Wenzlaff
3494 S Greenwood Ct
Eagan MN 55122

I recently purchased one of the PolyMorphic Video Terminal Interface units from a local computer store. After opening the plastic bags included with the kit and checking the parts against the packing list, I sat down to the task of assembling the kit. The instructions looked simple enough, the parts were all there, and there was a parts diagram, except I couldn't read it.

Well, being no stranger to electronics, I armed myself with a pen, the schematic, and a bottle of Dr Pepper. (The Dr Pepper is important!) Some four hours and many bottles later, my board was complete. Tracing circuit diagrams is OK if you have a lot of time and know your electronics, but there have to be a lot of nonelectronics people who bought this board and had the same problem. The more I thought about it, the more curious I became. A call to PolyMorphics gave me the answer.

It seems that some of the first instruction books printed managed to get by the quality control department without being checked. In any case, I spoke to a very nice person by the name of Cindy Feeney, who turned out to be the national marketing manager. She apologized for the problem, and explained that they sent a letter to their dealers as soon as they became aware of the mistake. The only trouble is they *didn't* know who had purchased the boards with the bad diagram, so some of us unfortunates got hold of a kit without knowing about the letter. She explained that the diagram had been reprinted, this time in three colors for easy readability. And she sent one to me. Free! She also offered to replace the diagram to anyone else who has had the same problem if they will just drop her a note. The address is PolyMorphics, 737 S Kellog Av, Goleta CA 93017.

Now I needed a television to connect it to. I don't own a video monitor, but I do have a black and white television set, a Panasonic Model TR-542. With this set, adding a video input is a cinch. A schematic of the section of the video amplifier to be modified is shown in figure 1. Panasonic provides a test point at the input of the video amplifier. The signal is positive going (signal is positive with respect to ground) and the level is 0.9 V peak to peak. The PolyMorphics board provides a positive going video output, with about 1 V peak to peak level. Talk about a perfect match!

In order to eliminate any biasing problems for the video amplifier, I elected to leave the video intermediate frequency (IF) stage connected. The PolyMorphics board has plenty of video to drive the amplifier, so the only thing you need to do is turn the television to an unused channel, preferably

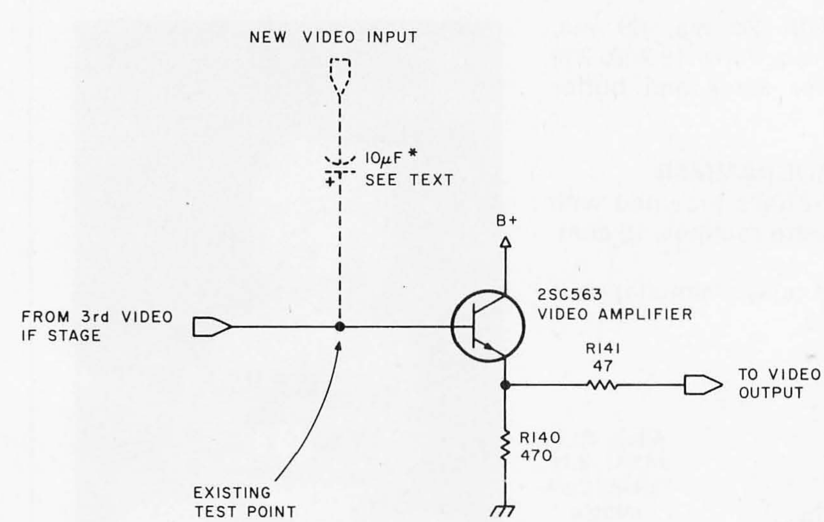


Figure 1: This is the simplified schematic for the video amplifier. The dotted capacitor is the added component to the original circuit. Take care to connect the positive side of the capacitor to the television circuitry. The video signal from the video interface can be connected directly through this capacitor.

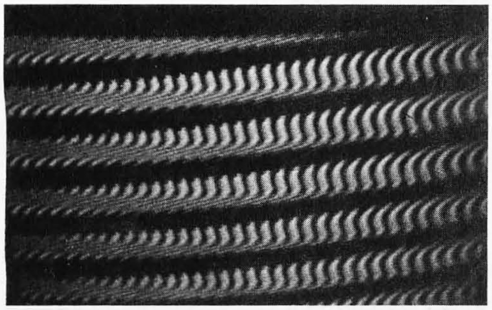


Photo 1: This is the display generated by the program written to output a series of vertical bars. The dark bands running across the monitor indicate that the horizontal frequency of the video interface and the horizontal frequency of the monitor are not identical.

UHF. The PolyMorphics board provides a slight amount of DC bias on the video line, and this will distort your display unless you filter it out. The simplest cure for this is the addition of a coupling capacitor in the video line. Install it in series with the center conductor, with the positive end connected to the TV circuitry as shown in figure 1. Experiment to see what value works best for you, but it will probably be between 0.1 and 10 mF.

With the television modified and the board completely assembled, I was now up to the section labeled 1.6. For those of you who don't own the PolyMorphics terminal, that section reads, "As it stands now, your unit should work if connected via coaxial cable to a video monitor or modified television set." Wrong. Not that there's anything wrong with the terminal, but I have built two of them now, and they don't do anything until you put something in memory. For ease of testing, the following program can be entered via the front panel switches. This eliminates the need for anything but a computer, the video interface board and the modified television. Set the PolyMorphics address to 0000, then proceed as follows:

```
0000  21
0001  0A
0002  00
0003  3E
0004  38
0005  77
0006  23
0007  C3
0008  03
0009  00
```

This program should display alternate black and white vertical bars on the screen. I say "should" because the display I got is illustrated in photo 1.

Notice the dark lines running from the lower left to upper right hand corner of the screen. These are present because the horizontal frequency of the set is not the same as the horizontal frequency of the video terminal. The standard horizontal frequency of a television set is 15,750 Hz. The frequency of the PolyMorphic board is 17,094 Hz. Although the manual would lead you to believe that this is a simple adjustment of the horizontal hold control, not all sets can be adjusted to operate at this frequency without some internal modification. I had access to four standard video monitors which I later tried the board on. Only two of the four were able to lock in and produce a usable display. Fortunately it's not too

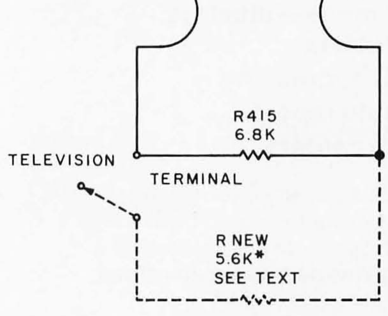
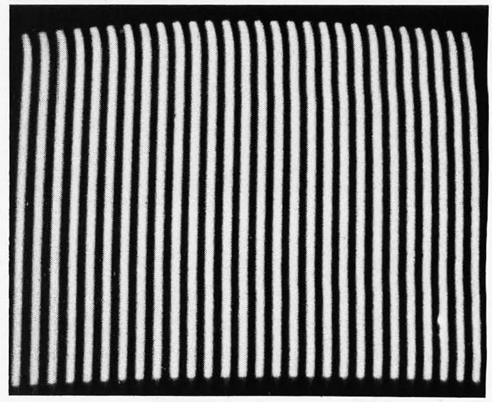


Figure 2: The section of the horizontal hold circuit that determines what the horizontal frequency is must be modified to match the horizontal frequency of the video interface. The dotted resistor and switch are additions to the original circuit. By changing the value of R415, I changed the range of the horizontal frequency adjustment. The switch is not necessary but allows the use of the set as either a monitor or a television.

Photo 2: The correct display should have vertically aligned black and white bands. This indicates that the horizontal frequency of the video interface and the monitor are identical.



difficult to modify the set once you know what you are up against.

The sets that seem to cause the trouble are the ones that use a coil to adjust the horizontal frequency instead of a potentiometer. A quick look at my Panasonic located the culprit. The horizontal frequency is determined by a coil, in conjunction with a 6.8 k resistor (R415) as shown in figure 2. Although it would be difficult to change the coil, we can adjust its range by changing R415. A 5.6 k resistor added in parallel with R415 changed the range sufficiently to produce a proper display. I could have permanently altered the value of R415, but then the set would not have been usable as a standard TV set. By connecting a switch in series with one of the leads from the 5.6 k resistor I can disconnect the added resistor from the circuit when I'm not using the set as a monitor.

The final display is shown in photo 2. Adding a video input required only one part, a capacitor. Correcting for the unusual horizontal frequency took a single resistor. This may not work for every set, but you'd be surprised how many sets use a circuit very similar to this one. If you have a tight budget, arm yourself with a schematic of your TV, a few spare parts, and this article. You'll undoubtedly learn something, and the pleasure of doing it yourself can't be beat. Besides, assuming you already own a black and white television set, where else can you get a video monitor conversion for under a dollar?■

Continued from page 91

"Start the program."

Time passed, a lot of time. Jack stabbed the RESET button hard enough to push the computer across the desk.

"Gently, Jack! I get your message. You must be putting the breakpoint in the wrong place."

"If I knew where to put the breakpoint, then I probably wouldn't need one. What I need is some way to sprinkle a program with breakpoints and just skip the ones I don't need."

"No can do, Jack. My MIKBUG monitor traps every breakpoint and that is that. You can't skip by one. If you put obstacles in my path, I trip over them. You don't want a bruised computer, do you, Jack?"

"I guess not. What I do want is a better way to debug. There's got to be something more effective than this 'stab in the dark' approach."

"May I make a suggestion, Jack?"

"Now look who's the designer. What words of wisdom have you, O great sage of Motorola?"

"Sarcasm will get you nowhere, except maybe 'stabbed in the dark.' I was going to suggest that you investigate my HALT input. If you put a properly timed signal there, then I'll execute only one instruction at a time. You can run programs so slowly even a human can follow the processing."

"That's an interesting idea. Let me think about it for a while."

"I can hardly stop you, Jack. I don't have hands. . .yet. You were looking at those robot articles in BYTE, weren't you!"

"Talking is quite enough, computer!"

"I. . .guess. . .so."

Jack sat back in his chair and thought. Computer knew better than to interrupt such meditations of his human partner. Computer liked its power continuous.

"No good, computer." Jack rolled his chair to the console again. "Hardware single stepping isn't what I need. I need to be able to read your registers and check memory locations. In short, I need your MIKBUG capabilities to help me debug. With your hardware suggestion I'd still need to know where to stop single stepping. That's no better than breakpointing."

"Not exactly, Jack. If you don't muck up my contents with your debugging stuff, then you *can* resume running again after you stop stepping. You can write reentrant code, can't you, Jack?"

"That's exactly what I'm trying to debug. Thanks a bunch."

"Sorry. I guess we'll both have to live with MIKBUG for a while longer, until you write me a real nice monitor, with asynchronous IO, and disks, and. . ."

"Get off the disk kick. A debugger is what I need. I want a purely software answer. I need to have MIKBUG-like facilities that I can use wherever I want in a program without upsetting that program. It's got to be reentrant. It's got to know how to break down instructions. It should give me a sort of breakpoint for each instruction executed."

"The program you seek is called a tracer. They're available on big machines, like your partner Grappel's PDP-11. Maybe he can adapt one to your liking."

"And adapt it to your limited faculties."

"His big machine can't even talk! Don't you say I'm limited!"

"Okay, okay, I give up. Anyway, it's bedtime. Good night."

"Yeah," said computer. Jack flipped the power switch, and computer's red eye dimmed.

. . . .

"So what's new?" said computer as its fan began to hum.

"Well, I uh. . .found. . .discovered that . . .noticed, uh. . ."

"Come on, Jack, out with it!"

"That problem you were having yesterday. . ."

"I wasn't having any problem yesterday! It was your code that was a problem. I just read 'em; I don't write 'em!"

"I know. But you should have warned me that I was pushing one more item onto the stack than I was popping off. When you executed the subroutine return, you got a byte of data confused with the real return address."

"I did not confuse anything! I did exactly, I repeat, exactly, what you asked for. You said PSH, I pushed! You said PUL, I pulled a byte off the stack. You said RTS, and I took the top of the stack as a return address. I may have bugs in the program, but the programmer's got bats in his belfry! If you can't count the number of bytes you put on the stack, you might think of going back to philosophy!"

"Cool it!"

"I might say. . ."

"Cool it!"

Jack glared at the console, and computer's red eye stared back. "I'm sorry, Jack."

"I guess it really is my fault, computer."

"Friends?"

"Friends."

"Going to get a tracer written?"

"Yep."

"Can I assemble it? I'll do a very careful job."

"I'm sure you will, computer. I'm sure you will."

. . . .

"Computer, let's try to work this breakpoint thing out."

"Glad to help, Jack."

"Fine. Now, we need a program which doesn't change any register or condition code or memory location in the target program. . .the one I need to debug."

"It's got to be reentrant. Right, Jack?"

"It should print the contents of all your registers, the address of the present instruction, and the instruction code. Something like the MIKBUG format should do."

"That's a problem. How do I do all that printing without messing up the registers?"

"Come on, computer. . .that's easy. You save all the registers before printing and then restore them when you're done."

"Like the MIKBUG software interrupt does, on the stack! You know, sometimes you're pretty smart, Jack."

"Except we can't do it that way."

"Why?"

"Because MIKBUG won't let me change the address of the software interrupt handler program. It's in ROM, unfortunately. We'll need another way."

"Jack, isn't this breakpoint thing sort of like a subroutine? I mean, it's, say, 'called' from the target program. . .does some stuff like printing. . .and then returns to the target program."

"I guess we have to do it that way. We'll put a subroutine call (JSR) at the address where the trace is to begin. It will call the trace program, which will be written as a subroutine. The subroutine will first have to save all the registers, then print my debugging info. It can then restore the registers and return. Thanks for the idea, computer."

"Don't thank me yet; it won't work. If I insert a 3 byte subroutine jump into the target program, then I've destroyed three bytes of your code. Then, when I return from the subroutine, I return three bytes further into the target program, not where I started."

Jack thought a bit and puffed his cigar.

"Jack! That cigar smoke is getting in my cassettes! How can you humans stand all that stuff? Do computers get cancer of the integrated circuit or something?"

"Relax, my automated friend. You're quite safe. I just figured out how to work the tracing."

"I'm all ears."

"I'm surprised you can stop talking long enough to listen. Anyway, I can overcome your objections by careful programming. Before inserting the subroutine jump, you'll save the three bytes you're replacing. You can put them back before you return."

"But, Jack, I still return to the wrong place!"

"Hold it a minute! I can fix up the return address on your stack to back it up three bytes. Then you'll return to the code you've replaced and restored. That'll be a breakpoint that I can really use."

"Glad to help you. But, Jack, you still have to know where to breakpoint. We're scarcely better off than we were with MIKBUG. True, the program can now continue after your breakpoint. Is that all you wanted?"

"It's enough for right now, but we'll probably extend it later. Please assemble this code." Jack placed a cassette in the drive and pressed PLAY. Jack smiled. "It's the only sure way to keep it quiet."

. . . .

All a trace is is a moving breakpoint. . . You handle it in the usual way except that before you return you put a new breakpoint where the next instruction will be, and fix up the previous breakpoint as if it were never there.

Why anybody would try to trace a program with interrupts going off is beyond me, but we'd better be complete.

"Computer, I want to extend Bob's breakpoint."

"It was only a matter of time. I suppose you want a full trace now."

"Right. It isn't that much more. All a trace is is a moving breakpoint."

"If you can't figure out where you want your breakpoint, then you make me push it around through your stuff. Why is it that I always have to bail you out of your problems?"

"That's what I built you for, remember?"

"Calm down, Jack. I was only kidding."

"I didn't build your sense of humor, that's for sure! Anyway, here's how you'll trace a program. Start with a breakpoint. You handle it in the usual way, except that before you return you put a new breakpoint where the next instruction will be. Effectively, this breakpoints every instruction!"

"Some things are easy to state in words but hard to code. How do I figure out where my next instruction is? I have instructions of different lengths in my op code set. I might jump or branch. . ."

"Computer, remember the 'Thompson Lister' program on page 99 of the October 1976 BYTE? It could figure out how long an instruction was by disassembling your code in memory. Well, I'm going to give you a version of that algorithm so that you can find the next op code. It'll also help you format the instruction printout for my ease in reading."

"Fine. . .if you think you're up to it. Besides, I remember that the 'Thompson Lister' couldn't catch invalid instructions. Sometimes you stick data into a subroutine return address and force me into the middle of nowhere!"

"I remember that incident well enough. I'll add a table of invalid op codes so that you can call me names when you hit one."

"This I like."

"I thought you would. Now, think you can trace?"

Computer sat with lights quivering. "I've got problems, Jack. You've given me a way to find the next instruction in most cases, but what about jumps or branches? Knowing the length of the instruction is no help."

"True. I guess we'll need a set of special cases."

"Oh boy. Here we go."

"It won't be too bad." Jack didn't sound too convinced. "Let's start with the jumps. There are subroutine jumps and unconditional jumps. They can be indexed or extended addressing."

"The subroutine stuff doesn't matter, Jack. For my purposes, a jump is a jump. All I need is the location of the end of the jump."

"Fine. So, we'll have two special cases: extended jumps and indexed jumps. The extended jumps are easy; the second and third byte of the instruction are the address you require to set your new breakpoint."

"Done."

"The indexed jumps need the contents of the index register from the target program, but you have saved that! You have the offset in the second byte of the instruction! Do a simple addition and you have the new breakpoint address!"

"It's simple if you give me a 16 bit addition program."

"Surely. Now for subroutine returns. You can get the return address from the stack. You've saved the target program stack pointer, so you can get the top of the target stack for your new breakpoint. That's special case 3."

"But what about all the branches?"

"That will take a bit of work. Let's work on the unconditional branches first; they're simpler. You do know where the target program is because you've got its program counter saved. You get the offset from the second byte of the instruction. You just add the offset to the program counter."

"What about signs, Jack?"

"Oh, yes. Forgot about that."

"I noticed that."

"All right, computer. You get a gold star! If the offset is negative, you must subtract it from the program counter. I'll give you a 16 bit subtract too."

"All that for just unconditional branches! I shudder to think what the conditional branches will need."

"Not too much more. We just have to decide whether the branch will be executed or not. If not, then the branch is just another 2 byte instruction. If it is to be executed, then it is equivalent, for your purposes, to an unconditional branch. You've already got code to handle each case."

"Yeah, but how do I know if the branch is to be executed? ESP?"

"Nothing but good, clever programming is needed here. You have the condition codes from the target program saved away. You have the op code, the type of branch. All it takes is a little trick. You'll copy the branch into a spot in the trace code and set the condition codes from your save area. Then, if the branch falls through, you know to treat it as a normal 2 byte instruction. The branch will tell you when to use your branch code. Simple, huh?"

"Self-modifying code. . .very poor form, Jack!"

"Can you do it better?"

"No."

"Then stop complaining. It's effective;

it works. Don't knock it."

"At least it will have your name on it and not mine. Any more special cases?"

"A few. We've got to take care of the interrupt instructions RTI and WAI and SWI. Why anybody would try to trace a program with interrupts going off is beyond me, but we'd better be complete. They won't be hard to handle."

"Thank God!"

"Since when did you get religious? Anyhow, the RTI is just like the subroutine return; just the return address is deeper on the stack."

"That was relatively painless. I can figure out the SWI code myself. I know the software interrupt will get a handler address from its vector, which, since I have MIKBUG, is in ROM. My new breakpoint goes at the address found in the vector."

"Very good, computer. Now, the WAI is a bit of a problem. You can't know whether the interrupt that will get you out of wait state will be an IRQ or an NMI. They have different vectors. We'll just have to pick one and warn the user of my tracer that the other type of interrupt causes problems."

"The IRQ is used more often, so I guess I'll get my new address from the IRQ vector."

"I guess that's a good choice."

"Done with special cases, Jack?"

"I think so. Here, I'll load this program and you try to trace it."

Computer began to trace. Jack smiled as the printout overflowed down the printer. Suddenly, the printing stopped. Jack punched RESET.

"I was going good there, wasn't I, Jack?"

"Yeah, but why did you stop?"

"You had this call to MIKBUG in the target program. I traced the next instruction and put my breakpoint out, but then everything fell apart."

"Of course, of course! You can't put breakpoints into ROM! You can try to store anything you want, the data won't change! When you breakpoint, check that your breakpoint is going in. If not, quit before you get lost in thought."

"Now you tell me!"

"Better late than never. Now let's see, we can't trace through ROM or nonexistent memory and we can't tolerate nonmasked interrupts at all, or IRQs unless we were in a wait for interrupt state. Can you think of any other places we'd have trouble?"

"Well, if you hit my RESET then I'll have trouble. I might not have fixed up my breakpoint yet."

"Right. Tell you what: every time you fix up the code after having traced an instruction, wait for me to hit a key on the console. This will let me stop tracing cleanly."

"Glad to oblige. Now, your favorite trick of modifying instructions could cause problems. If an instruction tries to modify the instruction I've tried to breakpoint, well, kaboom!!!"

"Very graphic."

"You're buying me some graphics equipment?"

"No, my eager processor. Perhaps a muzzle. . ."

"Okay. Beware of tracing programs which use modifying instructions. You shouldn't write them that way anyhow."

"Computer, try tracing this now."

The stream of printout began again, with Jack periodically tapping the carriage return key. "Wait a minute, wait a minute! Computer, you're getting some of these branches screwed up."

"I'm just doing what you said to do."

"Well maybe I was wrong."

"Please publish that last comment, Jack! I want that admission in writing!"

"Okay. Now, what's the problem? Why do some branches trace properly and others don't?" Jack poured over the printout while computer hummed contentedly.

"Bob! Come here and look at this!" (Enter Bob, who really was there all the time, but didn't say much.) Bob scanned the trace listing.

"You always get forward branches right. That must be a clue. What is it about backward branches? You get some of them right." Bob thought some more.

"Oh, sure!" Bob jumped to the console again, papers falling to the floor. "If you branch backwards less than three bytes, then your new breakpoint overlaps the present instruction!"

"Fine, Bob. Now what are we to do about that? My breakpoint has to be three bytes long."

"Yes, but this problem only happens on backwards branches. A branch doesn't change anything in the target program except the program counter. In fact, it needn't be executed at all. We just change the return address from the trace routine to get back to the right place in the target program! We return to the breakpoint call, not the branch! It's easy."

"Fine, Bob. Can I rest now? It's been a long time since I had some time to myself. All work and no play makes Jack's computer dull."

. . . .

"Computer!"

"What is it, Jack? I was just reading that new language you guys have been working

Of course, you can't expect to put breakpoints into read only memories. . .

on, STRUBAL. Bob wants me to compile it for him. It looks like a big project."

"Well, right now I want you to help me extend our debugger."

"You never give up, do you, Jack?"

"With such an able assistant, why should I?"

"That's hitting below the belt."

"You don't have a belt, computer."

"I forgot," said computer sheepishly. "What now?"

"Your tracing is very helpful, but I'd like to be able to fix the errors that I find without reloading the program and retracing my steps."

"Would you say 'our steps'?"

"If you insist."

"I do."

"Okay. We don't want to retrace our steps. We need more of MIKBUG's capabilities in the debugger. I want to be able to change the register contents in the target program."

"After I spend so much effort saving the contents?"

"Yes. If I find that a register has the wrong thing in it, then I'll want to correct the register before you go on to the next instruction."

"Well, that's no big deal. I just change my stored value for that register. Then, when I return to the target program, the register will have what you want in it. How will you tell me which register to change?"

"I thought a lot about that, and I think I will use the console input that now tells you to go on. From now on, if I type a carriage return, then go to the next instruction. If I type a capital A, then I want to change your A register. If I type a capital B, then I want to change your B register. Similarly, X and S indicate your index and stack registers. Just after the input you can wait for me to type in the new value I want in that register."

"I suppose I keep letting you change registers until you get around to a carriage return?"

"Right, and, if I type something that doesn't correspond to a register, just skip it. Prompt me for another input."

"Yes sir, boss. Let me anticipate your next request. You want to be able to change memory locations, like MIKBUG does."

"Right again! We'll indicate that with a capital M. I'll enter the address. You give me the present contents and then let me type my desired value for that location."

"Done. I'm going to add a feature that might be useful. I'll automatically convert lower case letters to upper case. Then you won't have to worry about case shifting on that fancy console."

"That's a good idea. Thanks."

"Glad to help. At least it will keep the swearing down when you forget to shift."

"Yes."

"Jack, I've got a question."

"What?"

"If you can change registers and memory at will, can't you get me into situations where I can't continue a trace? Especially if you muck around with the stack."

"I guess that's true, but let the user beware. I don't expect you to protect against every stupidity that a programmer may come up with. All the legitimate cases I can think of will work correctly. After all, the trace program is only about one kilobyte."

"I'm glad you said that and not me."

"Computer, we understand each other."

"Yeah, Jack. Now can I go back to reading STRUBAL?"

"I suppose so."

"Jack, would you put a clean cassette in drive 1? I think I may be needing it."

"Sometimes I wonder who works for whom," muttered Jack as he reached for the bulk eraser. He dropped the cassette

into the drive. It began to slowly and inexorably turn.

. . . .

"Computer, load the tracer program, please."

"You want to change it *again*!"

"Don't get steamed up. I just want to run an example to test out the tracer."

"What target program should I load?"

"You don't need one."

"Come on, Jack, be serious. Of course I need a target program. You don't expect me to trace memory garbage. You don't mean that, do you, Jack?"

"You've already loaded a program; let's trace that."

"Trace the tracer. Clever! That will really show that tracing doesn't upset the target program. Okay, I'm ready."

"Go."

"What address in the program do you want to start at?"

"How about 212 hexadecimal?"

"212 it is. Here are your registers: index, condition code, B, A, and stack pointer. The instruction is a CLR B, hexadecimal 5F. What would you like?"

"Continue trace." Computer traced the next instruction. Jack typed a carriage return and computer traced again. Again Jack hit the return and computer traced. Jack hit yet another carriage return. Computer traced the instruction at 219.

"Why don't you show off some of your register change stuff? You're at a compare A with 8C immediate instruction; why not make A equal to 8C?"

"Fine. Do it."

"Done. What now?"

"Continue tracing."

"The tracing tracer traces, and having traced, moves on."

"Can the poetry and just trace the program, if you don't mind."

Computer traced the next ten instructions without comment. "Let's show some of the other debug stuff."

"Okay. Change the B register to FF."

"Done."

"Change the index register to 1234."

"Roger."

"Change the condition codes in the target program to D1."

"That's cute, Jack. What does it mean?"

"Just do it."

"All right. How about a memory change? I've got lots of memory that isn't being used right now."

"Fine. Look at location 500."

"It's got 22 in it now."

"Make that 44, computer."

"Your wish is my command."

"Continue the trace."

"I'm at 10B now. It's a jump to MIKBUG."

Jack hit a carriage return.

"Got to stop here, Jack. I can't trace ROM. Try a new address?"

"No, I think that will make a sufficient example." Jack turned and walked toward the kitchen. He almost imagined that he heard a sigh from the workshop. He ignored it.■

And when Tracer was done, Jack's computer sent his printer the following listing of tracer tracing tracer, ultimate confirmation of the program's operation. In this listing, the lines which are blank except for single colons illustrate inputs of carriage returns to cause the program to proceed with tracing the next instruction. Each line of output contains the hexadecimal contents of the index register, processor condition codes, B and A accumulators, stack pointer, current instruction address, and the current instruction's hexadecimal operation code and operands. After tracing through to location 0245, several memory manipulation and register manipulation commands are executed, followed by one further line of traced code.

```
ENTER START-TRACE ADDRESS: 0212
   X   CC  B   A   SP-ADDRESS   INSTRUCTION
:0212 D0 02 8C A042     0212    5F
:
:0212 D4 00 8C A042     0213    FE 0173
:
:0216 D0 00 8C A042     0216    A6 00
:
:0216 D8 00 A6 A042     0218    08
:
:0217 D8 00 A6 A042     0219    81 8C
:A 8C
:
:0217 D4 00 8C A042     021B    27 1C
:
:0217 D4 00 8C A042     0239    5C
:
:0217 D0 01 8C A042     023A    5C
:
:0217 D0 02 8C A042     023B    5C
:
:0217 D0 03 8C A042     023C    F7 0172
:
:0217 D0 03 8C A042     023F    5A
:
:0217 D0 02 8C A042     0240    27 09
:
:0217 D0 02 8C A042     0242    5A
:
:0217 D0 01 8C A042     0243    27 03
:
:0217 D0 01 8C A042     0245    BD 010B
:B FF
:X 1234
:C D1
```

Using Tracer 6800

A low level trace technique is a useful adjunct to an assembly language oriented program development situation. The tracer program described in this short story can be purchased by 6800 owners in the form of a Paperbytes^{tm} program product, number 2.1, soon to be published. *Tracer: A 6800 Debugging Program* includes a reprint of "Jack and the Machine Debug," tracer program notes, complete assembly and source listing, object program listing, and machine readable Paperbytes^{tm} bar codes for the object program. Watch BYTE for details on price and where to purchase *Tracer.*

Multiprogramming Simplified

Multiprogramming is the ability of the computer's operating system to handle and execute several programs concurrently. In this article, I've set out to explain in a simple fashion the concept of how the operating system of a computer handles more than one job (program) at one time. Only the essential elements are included in this simple model, which is based on a "typical" large scale computer's programming environment. The same general concepts are of course applicable as well to the much smaller memory regions of the typical personal computer.

The operating system, through its various control programs, keeps track of the amount and location of available memory and the specific memory regions allocated to programs currently in memory. As programs (tasks) are read into the computer, certain information associated with them is stored by the computer. The name of the program and the location where the above information about the program is stored is placed on a list called the job (task) queue (see figure 1). As memory becomes available, programs to be executed are loaded into memory (figures 2a and 2b) according to their size and arrival time (how long they have been waiting). Information regarding these programs, such as name and location, is placed on the ready queue. As processing continues, programs are categorized as either active, ready or waiting. Only one program at a time can be active.

Job Queue	Active Queue	Ready Queue	Waiting Queue	NSI Cells
A - 100 K				
B - 150 K				
C - 50 K				
D - 150 K				
E - 100 K				

Program Counter = xxxxxx

Figure 1: When a program is entered into the multiprogramming computer, it is first put onto a job queue. The jobs are typically stored in the order they are entered, and each queue entry has all the essential information about the job.

Job Queue	Active Queue	Ready Queue	Waiting Queue	NSI Cells
		A		A 100,000
		B		B 200,000
		C		C 350,000
D - 150 K				
E - 100 K				

Program Counter = xxxxxx

Figure 2a: Enough memory is available to fit the first three programs into the ready queue so they can await execution. The next sequential instruction (NSI) cell for each of the programs is initialized to the location of the first instruction in the corresponding program.

The operating system maintains a special memory location for each program in memory which contains the next sequential instruction (NSI) to be executed for that program. This memory location is called the NSI cell. As a program is loaded into memory, the address of the first instruction to be executed for that program is moved into this NSI cell. A special NSI register (program counter) is maintained by the hardware containing the address of the next sequential instruction to be executed for the currently active program. When a program becomes active, the next sequential instruction pointer is moved from its NSI cell to the program counter of the computer; this will of course be dynamically changing for the currently active program. As instructions for the active program are executed the value of the program counter is typically incremented by the length of the current instruction being executed to reflect the address of the next instruction address that is to be executed. When branches occur, the program counter is redefined completely. This process is repeated until the program is either completed or interrupted by an outside service request from a real time clock or IO operation. If the active program has been completed, its memory allocation is freed and becomes available for reallocation. If it was interrupted it will be

Irwin Lahasky
Bankers Trust Company
1 Bankers Trust Plaza
New York NY 10006

placed on the waiting queue and its next sequential instruction pointer will be defined by the old program counter value at the time of interrupt. The highest priority program in the ready queue will be given active status, its NSI cell will be moved to the program counter, and instruction execution will be resumed at its NSI address.

As IO requests are serviced, programs will be moved from the waiting queue to the ready queue, and will be returned to active status when their turn comes. Example: Program A (100 K bytes), program B (150 K bytes), program C (50 K bytes), program D (150 K bytes) and program E (100 K bytes) are read into the computer and placed on the job queue (figure 1). 350 K bytes of memory are available beginning at address location decimal 100,000. Addresses 0 thru 99,999 may contain operating system programs. Program A is loaded into locations 100,000 to 199,999, its NSI pointer is set to its first instruction to be executed (address 100,000), and it is placed in the ready queue. Program B is loaded into locations 200,000 to 349,999, its NSI pointer is set to its first instruction address of 200,000, and it is placed second in the ready queue. Program C is loaded into locations 350,000 to 399,999, its NSI pointer is set to 350,000, and it is placed third in the ready queue. 50 K bytes remain available in memory from addresses 400,000 to 449,999, but this is insufficient for either of the remaining programs (D and E), which require 150 K and 100 K bytes, respectively. Therefore this memory will remain temporarily unused (figures 2a and 2b).

If there is no entry in the active queue, the first program in the ready queue, program A, is moved to active status, its NSI cell is moved to the program counter (figure 3), and execution will begin at the status address. Program B now becomes first on the ready queue and program C second. As the instruction at location 100,000 is fetched and executed, the address in the program counter value changes as instructions are executed.

Assuming the first instruction is 2 bytes

Start	Contents	End
* Address 000,000	Operating System Programs	Address 99,999 *
* Address 100,000	Program A	Address 199,999 *
* Address 200,000	Program B	Address 349,999 *
* Address 350,000	Program C	Address 399,999 *
* Address 400,000	Unused	Address 449,999 *

Figure 2b: A representation of where the programs are actually stored in memory. Addresses 0 thru 99,999 (decimal notation) are used by the operating system of this example.

Job Queue	**Active Queue**	**Ready Queue**	**Waiting Queue**	**NSI Cells**
	A			A 100,000
		B		B 200,000
		C		C 350,000
D - 150 K				
E - 100 K				

Program Counter = 100000

Figure 3: Program A is moved into the active queue to be executed. The next sequential instruction pointer is moved from the appropriate NSI cell to the program counter upon activation.

Job Queue	**Active Queue**	**Ready Queue**	**Waiting Queue**	**NSI Cells**
	A			A 100,000
		B		B 200,000
		C		C 350,000
D - 150 K				
E - 100 K				

Program Counter = 100002

Figure 4: The program counter is here incremented by 2, since the first instruction of program A is a 2 byte instruction. This has no effect on the related NSI cell.

long, the next sequential instruction to be executed becomes 100,002 (figure 4). After the execution of the instruction at location 100,000 is completed, the instruction pointed to by the program counter (at location 100,002) is fetched for execution, and the program counter is changed to 100,004. This is done because the second instruction is also two bytes long.

Processing continues in this manner until an interrupt in processing is encountered, such as a request to read data into the program from an input device or a request to write data to an output device. In this case, time is required to get or write the external data, and control is transferred to another program in the following manner. For the purpose of our example, let us assume that program A issued a read instruction located at address 158,266, and that this instruction type is six bytes long. The program counter which had been pointing to address 158,266 will be incremented by 6 to 158,272 (figure 5). An interrupt is generated by this IO instruction.

The program counter contains the address where execution is to be resumed for program A (158,272). The program counter is stored in program A's NSI cell, and program A is placed last on the waiting queue. The next program in the ready queue is moved to active status, its NSI cell is moved to the program counter, and processing is continued at the address (now different) in the program counter (figure 6).

As jobs are completed and their memory allocation is freed, programs waiting on the job queue are loaded into available locations. Their NSI cells are initialized and they are placed last on the ready queue.

Programs are loaded into memory according to their position on the job queue, their memory needs, and availability of core. If program C, which occupies 50 K bytes, finishes first, its memory allocation of 50 K bytes plus the 50 K bytes which is unused (total 100 K bytes) is not sufficient for program D (which needs 150 K bytes) even though program D is next in line on the job queue. The 100 K bytes of available memory is sufficient for program E, so it is loaded into locations 350,000 thru 449,999. Its NSI cell is then set to 350,000 and it is placed last on the ready queue (figure 7).

This idea of multiprogramming has developed over a number of years of conventional computing systems, ranging from the simplicity of two interacting programs on small machines to the larger contexts of many jobs executing simultaneously on the biggest machines. It is an example of how creative programming and design of systems software can make a machine do more than what the hardware designer intended.■

Job Queue	Active Queue	Ready Queue	Waiting Queue	NSI Cells
			A	A 158,272
		B		B 200,000
		C		C 350,000
D - 150 K				
E - 100 K				

Program Counter = 158,272

Figure 5: When program A reaches memory location 158,266, read instruction is encountered that is 6 bytes long. The program counter is incremented by 6 to 158,272. While the read operation takes place, program A is removed from the active queue and put into the waiting queue. The current value of the program counter is then stored in the A NSI cell as shown here.

Job Queue	Active Queue	Ready Queue	Waiting Queue	NSI Cells
			A	A 158,272
	B			B 200,000
		C		C 350,000
D - 150 K				
E - 100 K				

Program Counter = 200,000

Figure 6: The next program on the ready queue is moved to the active queue after an interrupt. The appropriate NSI cell is moved to the program counter (re)starting the program on its way.

Job Queue	Active Queue	Ready Queue	Waiting Queue	NSI Cells
	A			A 158,272
		B		B 248,208
D - 150 K				
		E		E 350,000

Program Counter = 158,272

Figure 7: At this point program C has finished and has been removed from memory. There is not enough room for program D but there is for program E, which is loaded into memory and the ready queue. Program E's NSI cell is set to its first instruction's location. Program A's read operation has finished and program A is again in the active queue. The programs will continue to shift in and out of the active status as they are interrupted, until the entire series, and any that are read in later, is completed.

Comments on Paging Schemes

Just read "Give Your Micro a Megabyte" by R D Grappel in the July BYTE and I thought it was GREAT! I think I have a way to avoid the startup "difficulty" referred to:

1. Reset the "page written latch." This will keep the junk page from being written into bulk store.
2. Use a latch to disable the page comparator logic so as to force a "not equal" output. This will cause the page processor to begin the page fetch sequence. The comparator will be enabled upon completion of the fetch operation.
3. Use the output of the page comparator as the source for causing the main processor to wait. This will allow immediate response to a request for a nonresident page. The main thing, however, is that the page processor need not respond as quickly as before, since the comparator has already asserted the wait request to the main processor.
4. Update the page select register after the completion of the page fetch operation. By waiting until the newly requested page is in main store, the wait request will automatically be disabled at the proper time. Also, each time the update occurs, the comparator is enabled, thus the latch set during startup will be cleared to its normal run state.

It is clear that by doing things this way the startup sequence looks just like any other nonresident page fetch. ■

Technical Forum

James F Gentry
4116 Schalk Rd 1
Millers MD 21107

The disk system you want at a price you didn't expect from a company that understands systems.



THE VISTA $50 FLOPPY DISCOUNT



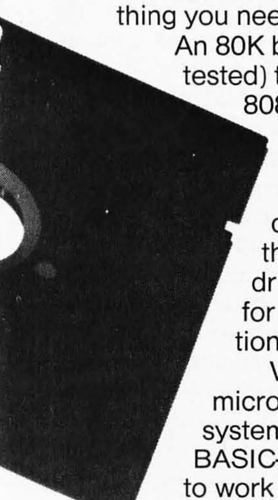
We know that one of the biggest problems in personal computing is that you're buying with your own personal dollars.

That's precisely why you're going to like doing business with us.

We're Vista Computer Company, the personal computer systems brainchild of the business computer systems people at Randal Data Systems.

And our V80 Floppy Disk System is a perfect example of how we're prepared to help you get the most out of your personal computing dollars.

$649 buys you the whole kit and kaboodle



The $649 you spend on a Vista V80 Floppy Disk System ($749 assembled) gets you everything you need:

An 80K byte minifloppy drive (assembled and tested) that can be powered directly by your 8080 or Z-80 computer. (Case and power supply optional.)

An I/O cable and a single card, S100 bus-compatible controller kit that handles up to four drives and includes a PROM for bootstrap loading (additional drives just $399).

VOS, the most advanced microcomputer disk operating system available, and our BASIC-E compiler, designed to work with VOS, all on a single diskette. Software functions include instantaneous program loading, named dynamic files, program editing, assembling, debugging, batch processing, and file copying on back-up diskettes.

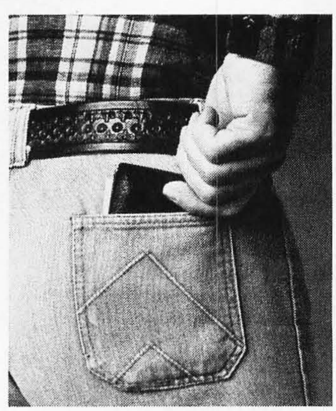
All backed by the Vista 90-day warranty, membership in VUE (Vista Users' Exchange), and Dataforce, our associated service company with 115 locations throughout the country.

Test drive the V80 at your local computer store

Drop by your nearest computer store and run the V80 through its paces. Once you find out what it can do for you, you'll see that our combination of high performance and low price is hard to beat and easy to take.

We love to take orders

If you'd like us to ship you a Vista V80 Floppy Disk System, they're available now. Just send us a check or money order for the amount of purchase, or your BankAmericard/VISA or Master Charge account number with expiration date and authorized signature. California residents add 6% sales tax. Uncertified checks require six weeks processing.



To place your order, or to obtain further information, call or write today.

Vista Computer Company, 2807 Oregon Court, Torrance, CA 90503. (213) 320-3880.

Vista

We never forget it's your pocket.

Book Reviews

Principles of Interactive Computer Graphics
by William M Newman and Robert F Sproull, McGraw-Hill, New York. $22.50.

It seems to be only a matter of time until someone somewhere develops a high resolution graphics display at a price within the reach of the amateur computer experimenter. Several companies have already introduced devices capable of generating medium resolution pictures using bus compatible cards and a common TV or monitor. Once the hardware is available, then comes the software: the programs that build the picture and move it in two or even what looks like three dimensions. A highly definitive and readable book by William Newman and Robert Sproull, *Principles of Interactive Computer Graphics,* leads the reader step-by-step through display devices, display files, interactive graphics, three-dimensional computer graphics and finally graphics systems. The book is designed as a college text, but this does not diminish its general usefulness. The language is not heavy and the math is for the most part limited to algebra. Nine appendices cover most of the nonalgebraic math and machine or language specific descriptions.

Part 1 describes available display devices including CRTs, storage tubes and plasma panels. Attention is given here for the nonhardware oriented reader to gain an understanding of what actually happens inside the equipment and what the advantages and disadvantages of each gadget are. Next follow two chapters on point plotting displays and vector generation. These are followed by a chapter on display processors, the highly necessary hardware that drives the actual picture making unit. Also covered here is an instruction set for a display processor that is used throughout the book.

Part 2 gets into the actual programming. For openers, the authors describe a hypothetical instruction set similar to that used on the IMLAC PDS-1 and the 18 bit DEC computers. This instruction set, along with that of the previous section and the SAIL language (an extension of ALGOL-60), is used in the numerous examples to illustrate sample techniques and methods. The authors then continue with a description of the use of subroutines and files to simplify and condense the instructions required to generate a picture. Finally, once you have built a picture, how do you move it? In the chapter

on transformations, both rotation about an axis and shifting along a line are discussed. Then comes the problem of eliminating part of a display that has been transformed off the screen. For the solutions the authors describe methods such as scissoring, clipping, windowing and viewporting.

Interactive graphics, part 3, is the real heart of the book and probably the most interesting section. Here is where graphic input devices such as the light pen or the Stanford Research Institute "mouse" are tied into the total system to provide a device which is efficient and usable. Also covered here are interrupt techniques, hardware versus software, and character recognition.

Three-dimensional graphics is given a good treatment with discussions of wire frame perspective, hidden line, hidden surface and shading, to name a few. This section once again discusses the transformation theory and some sample implementations required to move a three-dimensional object through space.

The last section, part 5, deals with needed languages as well as those now available for use in developing interactive graphics systems. Included are discussions of command languages, primitive operators, and some considerations on system design.

From a hobbyist standpoint, one of the best parts of the book is its extensive bibliography. Over 300 references are cited with a short preceding section which describes some of the most useful.

While it does not read like a popular novel, the book is exciting and tends to encourage "leafing through" until something interesting catches the eye for more detailed reading. The authors have gone to considerable effort to expound virtually every method or algorithm talked about with actual program segments and profuse illustrations (After all, what good is a book about graphics without lots of pictures?).

For the beginner in graphics who needs to know a little about everything or for the professional who needs a reference work, this book is a good place to start. The style is easy to read and skim for the novice, and the depth of the material presented is sufficient so that with the bibliography and appendices, even an old-timer can learn a lot.

If you are not afraid of something besides BASIC and the ASR-33 then this book can make a valuable addition to your reference shelf.

Steven Fox
875 Foxcroft Lp
Bosque Farms NM 87068■

Programming Quickies

A Number Guessing Game

Keith C Laudenslager
139 Bronx Dr
Cheektowaga NY 14227

This is a game to guess a number between 0 and 99 that the 8080 computer is thinking of. When the program if initialized, a "C" followed by a carriage return and line feed is sent out to your Teletypewriter or video monitor. To start, type in any 2 digit number using leading zero for numbers below ten. If it is not the number the computer is thinking of, the number will be typed out followed with an "H" or "L." The "H" or "L" indicates that you guessed either too high or too low. If the number is guessed correctly, it will be typed out and followed by the letter "C." To play again, just type in another 2 digit number. The program is listed in symbolic form along with absolute code, and was assembled by hand.■

```
1030 1E 43      START   MVI  E,43H    ;E:= "C"
1032 CD 86 10           CALL CHOUT    ;OUTPUT CHARACTER
1035 CD 91 10           CALL CRLF     ;OUTPUT CARRIAGE RET AND LINE FEED
1038 04         LOOP    INR  B        ;INCREMENT LOOP COUNT
1039 DB FE              IN   FEH      ;INPUT DATA
103B E6 04              ANI  04H      ;CHECK IF DATA AVAILABLE
103D CA 38 10           JZ   LOOP     ;LOOP UNTIL DATA READY
1040 78                 MOV  A,B      ;OTHERWISE GET COMPUTER NUMBER
1041 27                 DAA           ;ADJUST TO DECIMAL NUMBER
1042 47                 MOV  B,A      ;REG B:=NUMBER
1043 CD 9C 10   GUESSIN CALL INECHO   ;INPUT DATA AND ECHO
1046 CD AB 10           CALL ASCBIN   ;ASCII TO BINARY CONVERTOR
1049 0F                 RRC           ;MOVE FIRST
104A 0F                 RRC           ;DIGIT TO
104B 0F                 RRC           ;UPPER 4 BITS
104C 0F                 RRC           ;OF BINARY DATA
104D 4F                 MOV  C,A      ;REG C:=RESULT
104E CD 9C 10           CALL INECHO   ;GET SECOND DIGIT
1051 CD AB 10           CALL ASCBIN   ;ASCII TO BINARY CONVERTOR
1054 81                 ADD  C        ;PACK BOTH NUMBERS TOGETHER
1055 4F                 MOV  C,A      ;REG C:=BOTH NUMBERS
1056 3E 50      TEST    MVI  A,50H    ;REG A:= 50
1058 B8                 CMP  B        ;IS COMPUTER # ABOVE 50?
1059 FA 63 10           JM   ABOVE    ;GO TO ABOVE IF # ABOVE 50
105C B9                 CMP  C        ;NUMBER GUESSED ABOVE 50?
105D FA 6F 10           JM   NUMHI    ;IF ABOVE 50 GO TO NUMHI
1060 C3 67 10           JMP  CMPBC    ;GO TO CMPBC IF BOTH BELOW 50
1063 B9         ABOVE   CMP  C        ;IS NUMBER GUESSED ABOVE 50?
1064 F2 74 10           JP   NUMLO    ;IF NOT ABOVE 50 GO TO NUMLO
1067 79         CMPBC   MOV  A,C      ;ACC:=NUMBER GUESSED
1068 B8                 CMP  B        ;COMPARE COMPUTER AND GUESSED #
1069 CA 30 10           JZ   START    ;IF EQUAL GO TO START
106C FA 74 10           JM   NUMLO    ;GUESSED TOO LOW
106F 1E 48      NUMHI   MVI  E,48H    ;E:="H"
1071 C3 76 10           JMP  OT       ;OUTPUT "H"
1074 1E 4C      NUMLO   MVI  E,4CH    ;E:="L"
1076 CD 86 10   OT      CALL CHOUT    ;OUTPUT CHARACTER
1079 CD 91 10           CALL CRLF     ;OUTPUT CARRIAGE RET AND LINE FEED
107C DB FE      NXTNUM  IN   FE       ;INPUT DATA
107E E6 04              ANI  04H      ;IS DATA READY
1080 CA 7C 10           JZ   NXTNUM   ;LOOP UNTIL DATA READY
1083 C3 43 10           JMP  GUESSIN  ;PROCESS NEXT NUMBER
1086 DB FE      CHOUT   IN   FE       ;CHECK IF TRANSMIT
1088 E6 02              ANI  02       ;BUFFER IS EMPTY
108A CA 86 10           JZ   CHOUT    ;LOOP IF BUSY
108D 7B                 MOV  A,E      ;ACC:=DATA
108E D3 FF              OUT  FF       ;OUTPUT CHARACTER
1090 C9                 RET           ;RETURN
1091 1E 0D      CRLF    MVI  E,0DH    ;REG "E":=CARRIAGE RETURN
1093 CD 86 10           CALL CHOUT    ;OUTPUT CHARACTER
1096 1E 0A              MVI  E,0AH    ;REG "E":=LINE FEED
1098 CD 86 10           CALL CHOUT    ;OUTPUT CHARACTER
109B C9                 RET           ;RETURN
109C DB FE      INECHO  IN   FE       ;IS DATA AVAILABLE?
109E E6 04              ANI  04H      ;CHECK STATUS BIT
10A0 CA 9C 10           JZ   INECHO   ;LOOP UNTIL DATA AVAILABLE
10A3 DB FF              IN   FF       ;INPUT DATA
10A5 5F                 MOV  E,A      ;REG E:=DATA
10A6 CD 86 10           CALL CHOUT    ;ECHO DATA
10A9 7B                 MOV  A,E      ;RESTORE DATA TO ACCUMULATOR
10AA C9                 RET           ;RETURN
10AB D6 30      ASCBIN  SUI  30H      ;SUBTRACT ASCII ZERO
10AD FE 0A              CPI  0AH      ;IS IT A LETTER OR NUMBER?
10AF F8                 RM            ;RETURN IF IT IS A NUMBER
10B0 D6 07              SUI  07H      ;SUBTRACT ASCII SEVEN
10B2 C9                 RET           ;RETURN
```

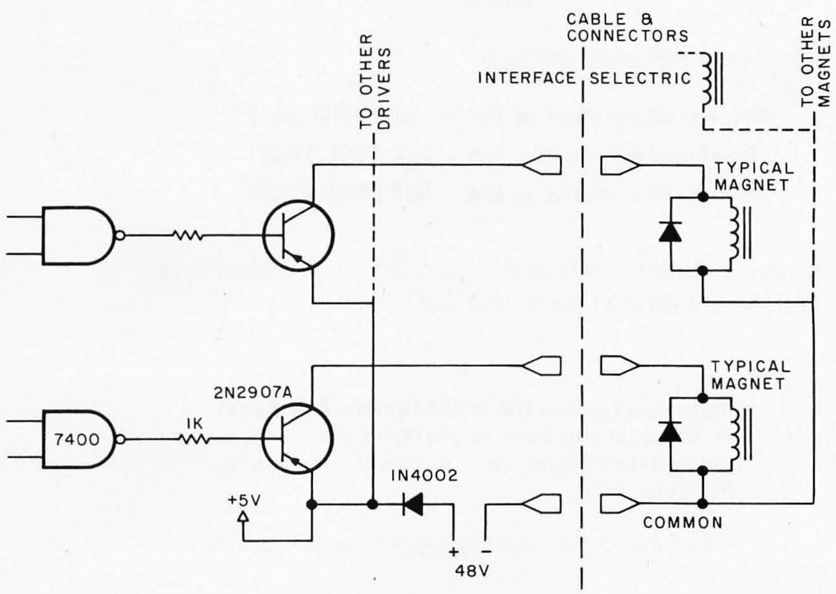
BYTE's Bits

TECHNICAL NOTE:
Selectric Interfacing Experiences

I enjoyed Dan Fylstra's well written article ("Interfacing the IBM Selectric Keyboard Printer," June 1977 BYTE, page 46), as his experience paralleled my own. Perhaps it is too obvious to mention but one big advantage of the Selectric is that it makes a fine stand alone typewriter. This is not impaired by connecting it to a computer and can help justify the outlay of several hundred dollars. There are many variations on the Selectric theme beyond the carriage size. Coil voltage may be 24 or 48 V, friction feed or a variety of traction feed (pin feed) platens are available, some printers are without keyboards, and Selectric mechanisms are used in other manufacturers' housings and equipment.

The IO Selectric (versus the typewriter version) is designed for continuous operation and once set up properly it should be quite reliable. Properly is the catch word since it is a complex mechanism and those who work on these machines professionally are used to getting professional prices. Surplus units can run the gamut from functionally perfect and well maintained to run-to-destruction and cannibalized hulks. I've been favorably impressed with the parts situation both in terms of price and availability.

For my Selectric output device, I also chose the open loop control system since it uses only one output port and less than 250 words of memory (8080 or Z-80). No interrupts or input ports are involved. With careful trimming of software timing loops very little speed loss should occur. Selectrics are more notable for quality print than speed; other impact printers are much faster. Mine runs about 12 to 14 characters a second, including carriage returns, and is nearly at the upper limit of speed.



A Note on Selectric Interfacing:
A Magnet Driver Circuit

I also started to do my interface with relays but hit on a simpler design. Mine has less isolation than Dan Fylstra's but this has been no problem. Driving either interface thru optoisolators would be good practice since relays and solenoids are very noisy. My actual circuit and program aren't given as I adapted a single relay to replace the missing dual latching shift relays. The front end is about the same except I used all eight bits. This will make it easier to actuate the check relay either with the tilt and rotate relays or as a control relay. The number of control

functions can be expanded with a BCD to decimal converter (7442 etc) between the 7400s and the transistors. For 48 V coils use 2N2907A (James, 5/$1) transistors and the suppression diodes should be on the relays. In this circuit the relay common line must not be grounded. Switching the motor on and off generates a big noise spike, worse on the inductive start motors than on the capacitor start ones. A zero crossing solid state relay would probably be the best solution and could allow TTL level on and off control.

What can you do with a printer? With a 64 or more character width monitor, floppy disk or computer controlled cassettes and a good text editor, you can easily store, retrieve, edit and print out letters, manuscripts, resumes, etc. I wrote this note on a Digital Group Z-80 system, stored it on a Phi-Deck, edited it on a DGSS text editor (awkward but well priced) and printed out the final version in 5 minutes. On a lesser scale, hardcopy of disassembled machine language programs or BASIC program listings make it much easier to see the whole program and structure or debug it. Printers are generally a poor substitute for a video monitor when running games. That approach can put you up to your armpits in paper. Paper that has been used on one side is often available free from commercial computer users and is fine (as is, or sheared) for listings, work sheets etc. This saves money and trees.

Don Southwick
7611 Aberdeen Way
Boulder CO 80301

IEEE

A call for papers has been issued for the Pattern Recognition and Image Processing Conference scheduled for Chicago IL on June 5 thru 7 1978. Sponsored by the Machine Intelligence and Pattern Analysis Technical Committee of the IEEE Computer Society, the program will consist of invited papers, panel discussions and contributed papers. Papers are invited on all aspects of pattern recognition and image processing, including statistical and syntactic pattern recognition, clustering, shape and texture recognition, scene segmentation and analysis, image filtering, enhancing and reconstruction, and medical, industrial and remote sensing applications.

There are two categories of contributed papers, and authors should state their preference when sending in material. Long papers (about 5000 words and suitable for a 25 minute oral presentation) are due December 1 1977. Short papers (about 500 words and suitable for a 15 minute oral presentation) are due February 1 1978. Camera ready copy in both categories is due April 1 1978.

Send image processing papers (in

triplicate) to Prof K Preston Jr, Department of Electrical Engineering, Carnegie-Mellon University, 5000 Forbes Av, Pittsburgh PA 15213, (412) 621-2600. All other papers should be sent (in triplicate) to Prof R L Kashyap, School of Electrical Engineering, Purdue University, West Lafayette IN 47907, (317) 493-9137.■

Siggraph 1978 Call for Papers

The 5th Annual Conference on Computer Graphics and Interactive Techniques will be held August 23 to 25 1978 in Atlanta GA, sponsored by ACM/Siggraph. The deadline for a 300 to 500 word abstract is December 15 1977. The first draft manuscript (including figures) is due January 16 1978. Send those to Prof R L Phillips, program chairman. 213 Aerospace Engineering Building, North Campus, University of Michigan, Ann Arbor MI 48109.■

Ambiguous BOMBacity

The following two comments were received, referencing the same article in the BOMB evaluations for May 1977. The author's name has been parameterized to "X" for purposes of anonymity. "Y" stands for the article name.

Comment 1:

<Y> is awful. <X> is an idiot. (Rating: 0)

Comment 2:

<X> is a great writer! He is clear, concise and yet detailed. (Rating: 10)

All one can conclude is that subjective evaluations of the same item sometimes differ markedly.■

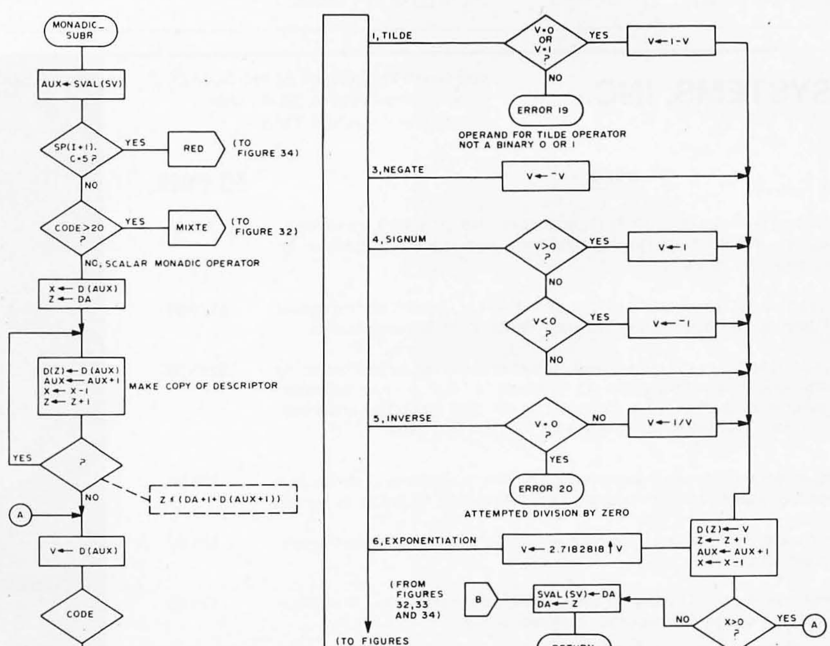
A Little Contest

A firm called Etronix has announced an applications contest featuring the use of Fairchild Technology Kits. The contest is open to anyone. It features a Fairchild Video Entertainment System ($169.95) as first prize; second prize is a Fairchild LED or LCD digital watch (approximately $80); third, fourth, and fifth prizes are Timeband digital LED watches (approximately $30). Send to Etronix, Box 321, Issaquah WA 98027, for your entry blank and rules. Contest closes December 30 1977.■

BYTE's Bugs

A Small Hole in the APL

There is one section of Mike Wimble's article, "An APL Interpreter for Microcomputers, Part 3" (October 1977 BYTE, page 64), that should be clarified. Figures 32 and 33 are actually continuations of the case structure labeled CODE and established in figure 31 (see page 66). A note should have been placed below the exponentiation branch at the bottom of figure 31 to make this clear.■



Clubs and Newsletters

Conducted by
David Wozmak

Eastern Iowa Computer Club

The Eastern Iowa Computer Club is a group of computer enthusiasts and other interested people. If you'd like to join, contact Samuel Dillon, 1125 Washington Dr, Marion IA 52302.

Illiana Teleprinter Society

The Illiana Teleprinter Society is a new Chicago based group of microcomputer experimenters. Meetings will be held on the second Thursday of each month at 8 PM in the lower lobby of the Clyde Savings and Loan Building, 722 W Cermak Rd, North Riverside IL. The programs will include speakers from Digital Equipment Corporation, IBM, and Heath Company, as well as an informal program of discussions and software swaps between members. All meetings are open to the public. Those curious about microcomputers and their uses are invited to attend. For more information contact John March, president, POB 874, Oak Park IL 60303.

Apple I Library

The Apple I software and hardware library is being started in Indiana to support the Apple I computer. Interested readers can obtain material at cost. Write to Joe Torxewski, 51625 Chestnut Rd, Granger IN 46530.

Jim's Industry Notes

This newsletter is aimed primarily at retailers and manufacturers. The newsletter provides fast turnaround news on industry data, and a communication medium for exchange of ideas, problems, solutions, etc. The newsletter contains no advertising.

Jim's Industry Notes is sent first class. Contact Jim Warren, POB 3010, Palo Alto CA 94305.

St Louis Amateur Computer Club

SLACC's newsletter, the *SLACC Stack*, contains club information and an applications forum. Meetings are held at 7 PM on the first Tuesday of every month at the Thornhill branch of the St Louis County Library. Contact Frank Curtis, c/o *SLACC Stack*, 24 Midpark, St Louis MO 63124.

CHIPS

A microcomputer club has been formed in the central New York State area. The club, known as CHIPS (Computer Hobbyists in Processing – Syracuse), has been holding regular monthly meetings and hardware and software demonstrations for almost a year. Membership is open to all who are interested in the microcomputer field. For further information, contact CHIPS, c/o J A Green, General Electric Company, Court St Plant #3, Room 16, POB 4840, Syracuse NY 13221.

Southern Nevada Personal Computing Society

The SNPCS meets on alternate Saturdays from noon to approximately 3 PM. Its membership is open to Clark County residents and students of Clark County educational institutions. Dues for a corporate member are $12 per year; family membership is $18 per year; correspondence membership is $6 per year; and student membership is $3 per year. Subscription to the newsletter *Hard Copy* is included in the membership. Meetings are held at Clark County Community College, Cheyenne Campus, room 3106. For more information contact Edna Wells (secretary/treasurer) at 1405 Lucilee St, Las Vegas NV 89101, (702) 642-0212.

Unofficial Heath Users' Group

A Heathkit computer users' group (as of this writing still unnamed) in New Haven CT produces a newsletter, *BUSS*, containing information about the Heathkit computers. You can get more information about the newsletter by writing to *BUSS*, c/o Charles Floto, 267 Willow St, New Haven CT 06511.

Microcomputer Tinkers and Bug Busters

This new club needs members and people willing to produce a newsletter. Contact Microcomputer Tinders and Bug Busters Society, 3845 Le Bleu St, Beaumont TX 77707.

Portland Computer Society

The Oregon based Portland Computer Society was formed in the summer of 1976 by Mike Enkelis and Mike Boyd when they discovered that the Altair computer had become a reality. To get in touch with the PCS, write to the Portland Computer Society, 3763 SE Division St, Portland OR 97202.

SEMCO

The Southeastern Michigan Computer Organization will be hosting the MACC Computerfest '78 in the Detroit Plaza Hotel on June 23 thru 25 1978. For club or Computerfest information contact SEMCO, POB 9578, North End Station, Detroit MI 48201.

A TRS-80 Users' Group

This group is dedicated to the exchange of programs and technical data for the new Z-80 based system by Radio Shack. Interested parties may send a self-addressed stamped envelope to R Gordon Lloyd, 7554 Southgate Rd, Fayetteville NC 28304.

Junior Computer Hackers of America

All students interested in forming a nationwide organization and publishing a newsletter of, by, and for computer oriented students should contact Brian Moran, 7335 N Manning Dr, Peoria IL 61614.■

Circle 2 on inquiry card.

Where to Get Bargains in Used Computer Equipment

Sol Libes, President
Amateur Computer Group of NJ
1776 Raritan Rd
Scotch Plains NJ 07076

Once they have their computers up and running, computer experimenters start hunting around for peripherals. They look for things like Teletypes (usually ASR-33s), video terminals, printers, paper tape readers and punches, etc. The big problem here is cost. In fact, those electronic and electromechanical IO gadgets can often cost several times the price of the processor itself. New video terminals can cost you $1500 and up. Printers can cost as much as $2000. What can a hobbyist do to save money? One way is to buy used equipment. The question is, where do you find it?

The big market in used computer equipment is due to many factors, one being that the state of the art is changing rapidly and companies frequently obsolete working equipment to keep up. Thus, there is a great deal of equipment available that may not be up to the latest speed or have the latest features, but is fully operational. For example, you can buy a used video terminal (ASR-33 compatible) for less than $500, or a video terminal that requires some rewiring for under $200. Hard copy terminals range from $300 (for an untested unit) to $1000 for guaranteed units with extra features.

There are even minicomputers available at bargain prices. Digital Equipment Corporation PDP-8s with 4 K words of core memory and serial interface start at $750 and go up to $3000 for newer models.

New dealers in used equipment are appearing all the time. The list that accompanies this report is not complete by any means, but it does include the larger dealers in the country. Most used equipment dealers publish catalogs and maintain mailing lists. A simple postcard will usually get you their latest equipment listing and put you on their mailing list. Most of these companies are eager to deal with computer experimenters.

Many dealers refurbish the used equipment they sell and restore it to manufacturer's specifications to the point where it is often indistinguishable from new equipment. For example, several dealers refurbish Teletypes to "as new" condition. This means a complete cleaning, replacement of defective or worn components, replacement of items such as plastic covers, repainting exposed metal enclosures, running the machine for at least 5 hours to insure its performance, and guaranteeing it for 90 days. A new Model ASR-33 Teletype

Used Computer Equipment Vendors	**Equipment**
American Used Computer Corporation POB 68 Kenmore Sta Boston MA 02215 (617) 261-1100	Used minicomputers, printers video terminals, modems, Teletypes, etc. Also new equipment for personal computing via the Computer Warehouse Store.
Atlantic Surplus Sales 3730 Nautilus Av Brooklyn NY 11224 (212) 372-0349	Teletypes, parts, supplies and miscellaneous.
Computer Warehouse Store 584 Commonwealth Av Boston MA (617) 261-2701	(See American Used Computer Corporation)
Data Access Systems 100 Route 46 Mountain Lakes NJ 07046 (201) 335-3322	Used Teletypes, DECwriters, video terminals, modems, parts.
Data-Lease 700 N Valley St #A Anaheim CA 92801	Minicomputers, video terminals, Teletypes, etc.
Data Processing Design Inc 6980 Aragon Cir, Suite B Buena Park CA 90620 (714) 994-4971.	PDP-11s, printers, terminals, etc.
Federal Communications Corporation Suite 107 11105 Shady Trl Dallas TX 75229 (214) 620-0644	Used Teletypes, data phones and TWX units
Herback & Rademan Inc 401 E Erie Av Philadelphia PA 19134 (215) 426-1700	Used computer gear and parts.

costs approximately $1200, while an "as new" reconditioned unit typically sells for $700 and an "as is" nonreconditioned machine goes for as little as $500.

Particularly good buys can be obtained on equipment no longer being manufactured or from a manufacturer who is no longer in business. Prices are often considerably less than those for equipment still being manufactured.

The largest and oldest used computer equipment dealer is American Used Computer Corporation, which was started in 1968. Second largest is Newman Computer Exchange, which was founded in 1972.

In addition to equipment dealers, used equipment is often advertised in publications such as *On-Line*, *Electronics News*, *Computerworld* and *Computer Hotline*. These should be consulted regularly for current "buys" and new companies in the market.

Another excellent source of used computer equipment is amateur radio flea markets. Ham radio operators are big users of Teletypes and other printers, tape readers and punches, etc. These flea markets are held locally and regionally. The largest is held in April in Dayton OH, and draws over 10,000 attendees. There are acres of flea marketers. To find out when and where the ham flea markets are being held, consult the ham magazines such as *QST*, *CQ* and *Ham Radio*.

Good hunting!■

Used Computer Equipment Vendors	Equipment
JM Associates 80 Emerald Av Westmont NJ 08108	Used computer peripherals, etc.
ICC Computer Corporation 1115 Security Dr Dallas TX 75247 (214) 630-1401	Used Teletypes and TWX units.
MiniComputer Exchange 154 San Lazaro Sunnyvale CA 94086 (408) 733-4400	Miniperipherals.
National Teletypewriter Corporation 207 Newtown Rd Plainview NY 11803 (516) 293-0444	Used Teletypes, parts and supplies.
Newman Computer Exchange 3960 Varsity Dr Ann Arbor MI 48104 (313) 994-3200	Used minicomputers, peripherals, Teletypes, etc.
PMR Canada Ltd 94 Hyde Av Toronto, Ontario CANADA (416) 653-4842	Used processors, memories, controllers, etc.
RCA Service Company Bldg 204-2 Camden NJ 08101 (609) 779-4129	Used Teletypes
Rondure Company 1224 Security Dr Dallas TX 75247 (214) 630-4621	Miniperipherals, modems, parts, etc.
Van't Slot Enterprises 550 Springfield Av Berkeley Heights NJ 07922 (201) 464-5310	Used Teletypes, parts and supplies.

A Look at LISP

Gary McGath
7 Silver Dr, #3
Nashua NH 03060

Today BASIC is the universal language of the microcomputer hobby. It is easy to learn, can run in a small amount of memory, and provides a common ground for people with different processors. But many microcomputers are no longer so "micro," and they need a more powerful language to use their full capabilities.

What properties should this language have? It should work well interactively, since most hobbyists use their machines that way. It should be simple in form, so that implementing and understanding it are easy. It should be good at handling nonnumeric data, since most computer hobbyists aren't interested in "computing" in the literal sense (ie: numerical calculations) as much as in graphics, information management and other such applications. It should run efficiently. And it should make programming easy.

A prime candidate, on all but one of these counts, is LISP. LISP was developed at MIT in the early 1960s; it has never attained widespread use, perhaps because of its unusual syntax, or else because it tends to run slowly. It is oriented not toward production work, but toward program development and experimentation.

Evaluating LISP in terms of the criteria mentioned above: It is very strong on interaction. Its syntax, though unusual, is very simple. It includes names and a versatile structured type in its data handling facilities. It is powerful enough to make most programming jobs simple. Unfortunately, though, it loses to many other languages in efficiency, at least when it is run by an interpreter (as is the normal case). But unless speed is so crucial for a given problem that it outweighs the convenience, LISP is an excellent choice.

The purpose of this article is not to be a primer on LISP, but to give you reasons to look for such a primer (for instance, Weissman's *Lisp 1.5 Primer*, published by Dickenson). There are, as far as I know, no microcomputer based interpreters for the language yet; but this could change, if suppliers see a potential market.

Much of the power of LISP comes from the fact that in it, programs are data. This means that one program can build up or look at another program, or even operate on itself. This is a tremendous help for debugging programs, since it lets them be examined or modified on the fly. For instance, a LISP routine might call a special error handling routine if it detected an unwarranted state of affairs; this routine could let the user examine and change variables, since variables in LISP are also data, or even change part of the program to get it back on the right track.

The structure which is used for programs and data is the *list*. A list is a sequence of any length, written as its elements enclosed in parentheses. For instance, (1 2 3 4) is a list of four numbers. Lists can contain lists; ((1 2 3 4)) is a list with one element, which is a list of four numbers. Note that in LISP, parentheses always change the content of expressions; they are never optional, as in some languages.

Programs can create lists as they run, thus giving themselves the storage they need. This is another speciality of LISP, called *dynamic storage allocation.* Most programs have to set out their storage requirements before they are run; a BASIC program, for instance, can't decide halfway through its execution that it needs another array. But LISP picks up its storage as it needs it. To give just one example of what this can mean: You can run a LISP program on a small amount of data with your current memory supply, then expand it when you get more memory to handle more data with no changes to the LISP program.

LISP programs are unusual, but consistent, in that they use one tool everywhere. This tool is *functional application.* BASIC programmers are familiar with simple functions like SIN; in the expression SIN (X), SIN is the function, which is applied to the argument X. In LISP, this expression would

be written as the list (SIN X). The rule is that the first thing in the list is the function, and the rest of the list is its arguments. A LISP program is a list of this form, which gets evaluated by applying the function to the arguments. The value of the list is the result of this application. This value may itself be used as an argument to another function; thus, programs can be built up from lists within lists to any level.

Some functions do other things besides returning a value. The function SETQ, for instance, performs the role of the assignment statement in other languages. (SETQ X 5) is like BASIC's LET X = 5. COND tests a condition and performs or omits evaluations depending on the truth or falsehood of that condition. This gives a capability like BASIC's IF statement, except that COND is much more general. The following is a simple example of how COND can be used:

```
(COND((GREATERP X Y) (SETQREL 1))
      ((LESSP X Y) (SETQ REL -1))
      (T (SETQ REL 0)))
```

This is what happens when the COND expression is evaluated: The function GREATERP is applied to X and Y. If X is greater than Y, the value returned will be T, otherwise it will be NIL. If T (or for that matter, any value but NIL) is returned, then (SETQ REL 1) will be evaluated, and the COND will be done. If, however, X is less than Y, then COND will try again on the first element of its next argument; that is, (LESSP X Y). If X is less than Y, this will evaluate to T, REL will be set to −1, and the COND will be done. If, however, this also falls through, then X must equal Y. T is then evaluated; since T is a special constant which always evaluates to itself, COND will finally be satisfied, and REL will be set to 0.

When a variable is used in a list, it is a piece of data like any other, of a type called *atom.* Atoms can be passed around by themselves (as opposed to passing around their values) by using the function QUOTE. (SETQ X Y) sets X's value to be the same as Y's value; but (SETQ X (QUOTE Y)) sets X's value to be the atom Y itself. This lets you keep verbal information around for later printing. QUOTE is also useful with lists, since it lets you create a list of data which isn't intended for evaluation.

The complement of QUOTE is EVAL. This takes an argument, already evaluated once by the application mechanism, and evaluates it again. The use of QUOTE and EVAL provides one way of writing a program for later repeated use; at some point you write:

(SETQ X (QUOTE *list-to-be-evaluated-later*))

Elsewhere you have:

(EVAL X)

When this is encountered, X is evaluated once by the LISP interpreter, giving *list-to-be-evaluated-later*; then EVAL evaluates it again, generating its value and all side effects.

A list can be modified (I'll discuss the methods when I get to the structure of lists), so program editing is possible. But simply evaluating fixed lists, even with editing, isn't a very flexible method of programming. It would be more useful to have a program that can operate on alternative sets of data; that is, to run it with one set of data, then run it again on different data. This can be done, of course, by SETQing variables on which the program will operate. But this

A sample LISP program for building a maze (which may be used in Wumpus type games). The user defines the maze by typing in lists of two atoms, which give two rooms that are adjacent to each other. The rooms may have as names any atoms not otherwise used. Input is terminated by entering NIL. The function returns a list of atoms which are the names of all the rooms; this value is also available as the value of the atom MAZELIST. Each of the room atoms has as its value a list of all the rooms to which it is adjacent.

The only function used here which was not mentioned in the main text of this article is MEMQ. This function tests whether its first argument is a member of its second argument, in the sense of list membership.

```
(SETQ MAZEBUILD
   (QUOTE (LAMBDA ()
      (SETQ MAZELIST NIL) ;initialize the value
      (PROG (TEMP) ;TEMP is a local variable
         A    ;A is a label
         (COND ((SETQ TEMP (READ)) ;get a pair
            (SETQ R1 (CAR TEMP))
            (SETQ R2 (CAR (CDR TEMP))
         ;the names of the two rooms
         (COND ((MEMQ R1 MAZELIST)
            ;is this room known already?
            (SETQ (EVAL R1)
               (CONS R2 (EVAL R1))))
               ((SETQ (EVAL R1)
               (CONS R2 NIL))
               ;if not, start it up
               (SETQ MAZELIST
               (CONS R1 MAZELIST))))
         (COND ((MEMQ R2 MAZELIST)
            ;do the same for R2
            (SETQ (EVAL R2)
               (CONS R1 (EVAL R2))))
               ((SETQ (EVAL R2)
               (CONS R1 NIL))
               (SETQ MAZELIST
               (CONS R2 MAZELIST))))
         (GO A) ;repeat if wasn't nil
         )))
      MAZELIST ))) ;return the value
```

means that each program must operate on different variables, or else bookkeeping problems start coming up. A more convenient method, which LISP provides, is to define a function which takes its data as arguments.

A user defined function is a list which has the atom LAMBDA as its first element. (This has its roots in a mathematical notation called the lambda-calculus.) The second element of this list is a list of atoms. These atoms are the parameters of the function, that is, the things that catch the arguments to the function. After the list of parameters, any number of expressions can follow.

The use of functions with parameters is familiar to FORTRAN programmers, and many versions of BASIC provide it in a simple form. What happens when a user defined function is called is this: First, the old values of the atoms on the parameter list are saved on a stack. Then the parameters are given the values of the corresponding arguments. (There must be one parameter for each argument.) Next, the remaining expressions of the function are evaluated in order. The value of the last expression is the value of the function. After this value is found, the old values of the atoms on the parameter list are restored from the stack. Thus, as far as the outside world is concerned, these atoms were never touched.

There are various methods for setting up function definitions in various implementations, and the method in Weissman's book is distinctly confusing. A straightforward method is to let the function be the value of an atom. Then to define a function, you would just have to enter

```
(SETQ FUN (QUOTE (LAMBDA (...) ...)))
```

At any later time you could simply use (FUN args) to apply the function.

There is, incidentally, no objection to using a function from inside itself. This method is called *recursion*, and it is often useful to break up a complicated case into simple ones. The idea is this: Your function examines its argument. If the argument is a simple one, it just returns the value. Otherwise, it finds a simpler case which has a known relation to the case at hand, and calls itself with an argument for the simpler case. For instance, the factorial function (FACT X), which returns X * (X – 1) *(X – 2) * . . . * 1, could be written:

```
(SETQ FACT
   (LAMBDA (X)
      (COND ((LESSP X 2) 1)
         ((* X (FACT (– X 1)))))))
```

(Note the use of arithmetic operators as

functions in this example.) The important thing to remember in recursion is that there always has to be a simple, nonrecursive case on which all the recursive cases are built; otherwise, you will find yourself blowing your stack without getting a result.

Input and output in LISP are simple. The function READ reads one LISP expression and returns that expression as the value of the function call; the function PRINT prints the value of its argument. Arguments to these two functions are necessarily implementation dependent; they may include the name of the device or some other indication of where to go for the IO function. A good system will also provide other functions, including ways of reading single characters and controlling output formatting. A desirable (but expensive) function would be one that printed a list in a readable indented format.

And now for the question that everyone has been asking: What are lists really like? A list really has only two parts, the first element and all the rest. These are called the *car* and the *cdr*, respectively, of the list. (Legend has it that these names come from the old IBM 704 implementation of LISP, where they stood for "contents of address of register" and "contents of decrement of register.") The *cdr* of a list is itself a list, containing all the elements but the first. In the case of a one element list, the *cdr* is the atom NIL which we met earlier in connection with COND. LISP takes NIL to be a list of zero elements; it can be treated as a list in most ways, and can be written as (); but, of course, it has no *car* or *cdr*. The functions CAR and CDR are available for taking the *car* and *cdr* of a list. There are also two functions, called RPLACA and RPLACD, for changing the *car* and *cdr* of an existing list.

This system lets you have pointers into any part or sublist of a list. For instance, if Y is some list, you might use

(SETQ X (CDR Y))

to get a version of Y without its first element. Both of these variables will then share the same storage; therefore, a RPLACA or RPLACD on X will change Y's value. The moral: Handle shared lists with care!

Unfortunately, the cost of the *car* and *cdr* approach is that indexing down a list is tedious. If you want the fifth element of a list, you have to work your way through four *cdr* pointers and then take the *car*. This isn't too bad when working linearly down a list, since a variable can be set to successive *cdrs* to get each element; but it does slow down random access.

The function which builds up lists is called CONS. CONS takes two arguments: The first becomes the *car* of the new list, and the second becomes the *cdr*. The arguments themselves are unchanged; what happens is that a new structure is created to encompass them both. This new structure is allocated from a pool of free storage. Using CONS is one of two ways to create new lists; the other is READ (which uses the equivalent of CONS internally), or the interpreter's top level reader.

But if you keep CONSing things, won't you eventually run out of free storage? Yes, this would be true if nothing were done to prevent this situation. But storage can often be reused. A list may be entered once and then forgotten; a variable may be reassigned, leaving its old value in limbo. This storage can be reclaimed by a routine called a *garbage collector*. Whenever the interpreter finds itself running out of storage, it can call the garbage collector, which finds storage that is available for reuse and designates it as such.

Garbage collection, unfortunately, is a slow process. Whenever it has to be done, the system comes to a complete halt for a few seconds. This can create problems for real time applications. I have heard reports, however, of a LISP system being developed which does its garbage collection in small chunks, so the waiting is distributed more evenly. Garbage collection is the price that has to be paid for dynamic storage allocation, unless you want to keep track of all allocated storage explicitly.

Now for the interpreter itself. Interactive LISP is run in a READ-EVAL-PRINT loop. That is, an expression is entered, it is evaluated, and its value is printed. The mechanisms used for these operations are identical to those of READ, EVAL and PRINT respectively. This means that a LISP function can start up its own equivalent of the interpreter anywhere the programmer chooses; this is often useful for debugging programs. You might, for instance, place a READ-EVAL-PRINT loop as the first thing in a function, giving it a provision for exiting, so that you could look at its parameters and decide if they are valid. This gives you an ability corresponding to breakpoints in machine language debugging.

But how do you do looping? The functions which allow this are PROG and GO. PROG constructs are similar to user defined functions, except that PROG is used in place of LAMBDA and the construct isn't applied to anything. The parameters are NIL, or else indeterminate, at first. The function GO can be used within a PROG construct, and

nowhere else. Its one argument is an atom; it treats that atom, which is not evaluated, as a statement label and goes to that label, if it exists in the PROG. For instance, a nonterminating READ-EVAL-PRINT loop could be written thus:

```
(PROG ()
   A
   (PRINT (EVAL (READ)))
   (GO A))
```

PROGs can put a GO inside a COND expression to allow for conditional branches.

Finally, LISP is an expandable language. Its costly features (in terms of designing the interpreter) are data definition, input and output, garbage collection, and a design that permits recursion. Once this exists, individual functions are fairly cheap. This means that with sufficient documentation, the user could add his or her own machine language functions to perform specialized operations (eg: graphics). LISP interpreters might even be sold in a minimal version, with add on modules available for larger memory sizes. (No, I can't say right now how much memory that minimal version would take.)

LISP is not an all-purpose language. If you like to do matrix inversions or quadratic interpolations, you will find it outrageously slow and awkward. But if you are interested in symbolic work, if you are not frightened by the unusual, if ease of use is more important to you than speed, and if your computer has the capacity, LISP may be your best choice.■

GLOSSARY

Atom: a variable or literal used in a list.

Car: the first element of a list.

Cdr: all elements of a list except the first element.

Dynamic storage allocation: the ability of a program to get the storage space it needs as it is executing.

Garbage collector: a routine that searches the storage space for previously used memory space that is no longer needed and allocates these memory nodes as free space.

Interpreter: a program that analyzes an instruction and executes it before going to the next instruction.

List: a data structure, represented as a sequence of elements enclosed in parentheses.

NIL: an atom that doubles as a list of length zero.

Recursion: the ability of a function to repeatedly call itself.

Relative Addressing for the 8080

James P Gaskell
Griffon Industries
Austin Rd
Amherst NH 03031

An essential characteristic of any computer is the ability to branch as a result of a decision. These jumps, or branches, can be done with target addresses determined in a number of ways. Examples include absolute (direct) jumps, indirect jumps, indexed jumps and relative jumps.

The instruction set for the 8080 processor includes explicit absolute jumps, both unconditional and conditional. It also has one indexed jump (although it allows for no offset). Unfortunately, the instruction set does not include any relative jump instructions, instructions which are necessary if position independent programming is to be accomplished. However, a routine can be used that simulates the desired result, thus enhancing the 8080's usefulness to programmers.

A relative jump goes to a specified offset from the present address. Instructions of this type allow jumps within a program to be independent of where the program is located in memory. That is, if the whole (object) program is moved to a new area of memory, the jumps are still valid. It is not necessary to reassemble or explicitly relocate a program every time it is moved around in memory.

In order to implement this routine, three steps have to be taken. First, a short program must be entered into memory. Second, a section of memory must be set to a constant. Third, some macroinstructions must be added to the 8080 assembler. (Macroinstructions are assembly language instructions that are expanded into a sequence of machine language instructions during assembly.) After these steps are taken, relative jumps can be written in a straightforward fashion in assembly language.

First, the program shown in listing 1 must be entered into memory. It is important that this program start at hexadecimal address

Hexadecimal Address	Hexadecimal Code	Label	Op	Operand	Commentary
0038	22 4B 00	ENTER:	SHLD	TEMP1	Save HL
003B	E1		POP	H	HL = (THERE − HERE + BIAS + 1)
003C	EB		XCHG		
003D	E3		XTHL		TOS = Original DE HL = (HERE + 3)
003E	F5		PUSH	PSW	Save PSW
003F	19		DAD	D	HL = (THERE + BIAS + 4)
0040	11 FC 3F		LXI	D, −(BIAS + 4)	
0043	19		DAD	D	HL = THERE
0044	F1		POP	PSW	Restore PSW
0045	E3		XTHL		TOS = THERE HL = Original DE
0046	EB		XCHG		Restore DE
0047	2A 4B 00		LHLD	TEMP1	Restore HL
004A	C9	EXIT:	RET		Jump THERE
004B	00	TEMP1:	NOP		2 bytes of temporary storage
004C	00	END:	NOP		

Note: In this listing, BIAS = C000 and TEMP1 = 004B.

Listing 1: A program used to perform relative jumps on the 8080 processor. A block of memory must also be set aside for the RST 7 instruction, and the macroinstructions listed in table 1 must be added to the assembler to allow this program to work.

0038, because it will be entered during execution via a ReSTart 7 instruction. Also, since TEMP1 is used for data storage, it is necessary that TEMP1 be located in programmable memory and not in read only memory.

Second, a block of memory must be set to the RST 7 instruction. The larger the block, the greater the range of the relative jump instruction. An address near the middle of this area should be chosen as the BIAS address. The machine language code for the RST 7 instruction is hexadecimal FF. This fortuitous coding permits an easy way to meet the present requirement. Select an area of (logical) memory where no hardware memory exists. If the data bus has pull up resistors, doing a memory read to these locations results in reading hexadecimal FF.

Third, macroinstructions should be written for the assembler so that it will do the computations necessary to implement the relative jump. Since relative jumps can be done either unconditionally or based on either state of any of the four flags, this requires that nine macroinstructions be written. Table 1 outlines what has to be done. If your assembler cannot handle macroinstructions, or if you do not have an assembler, you must enter the proper CALL instruction as shown in the table.

To help in understanding the theory of operation, refer to figure 1. Since we wish to jump from HERE to THERE, we insert JR THERE (= CALL (THERE - HERE + BIAS)) at HERE. This instruction pushes (HERE + 3) onto the stack and jumps to the location (THERE - HERE + BIAS). At this new location we have made sure that the contents are equal to hexadecimal FF (= RST 7). This instruction pushes (THERE - HERE + BIAS + 1) onto the stack and jumps to hexadecimal 0038. At 0038, we find our program. This program first saves the contents of selected registers. It then pops the top two words off the stack and adds them together to give (THERE + BIAS + 4). After (BIAS + 4) is subtracted off, we are left with THERE. This address is pushed onto the stack and the registers are restored to their original conditions. The final RETurn does a jump to THERE.

This routine uses four bytes in the stack and takes 78.5 μs (with a 2 MHz clock, ignoring memory wait times) to execute. For comparison, the absolute jump uses no stack locations and executes in 5.0 μs. This routine returns all registers in their original states (except, of course, PC). After the overhead is established, each relative jump requires three bytes, which is the same as the absolute jump. This means that switching between absolute

Macro Number	Macroinstruction	Assembled Instruction
1	JR LABEL	CALL (LABEL - $ + BIAS)
2 - 9	JR (FLAG) LABEL	C (FLAG) (LABEL - $ + BIAS)
Note 1:	$ means the location of the current instruction.	
Note 2:	(FLAG) means any of the eight condition flags, ie: NZ, Z, NC, C, PO, PE, P or M.	

Table 1: The nine macroinstructions which enable the assembler to perform relative jumps on the 8080 processor.

```
                .
                .
                .
                Instr
HERE:           JR THERE      ; CALL (THERE - HERE + BIAS)
                Instr                     |
                                     N locations
                Instr                     ↓
THERE:          Instr         ; Address = (HERE + N)
                Instr
                .
                .
                .
                .
                RST 7
BIAS:           RST 7         ; Contents = FF
                RST 7                     |
                                     N locations
                RST 7                     ↓
                RST 7         ; Address = (THERE - HERE + BIAS)
                RST 7
                .
                .
                .
```

Figure 1: An illustration of relative addressing as it is performed on the 8080 processor. The program jumps from HERE to THERE.

and relative jumps does not affect the length of the object code. As requirements change (basically, relocatability versus execution speed) it should not be difficult to alter the coding.

In summary, the ability to do relative jumps can be added to an 8080 based system. The working program requires only 21 bytes with perhaps another couple of hundred bytes located elsewhere, so the demand on memory space is quite low. In a typical application of relative jumps, object programs are pulled off a mass storage unit, placed in any available memory, and immediately used. This example shows some of the strength of this kind of jump and some of the reason for the experimenter to add this capability to his/her operating system.■

Save Software:

Use a UART for Serial IO

Fr Thomas McGahee
Don Bosco Tech
202 Union Av
Paterson NJ 07502

In my opinion suppliers of software for microprocessors should refrain from writing their software for serial IO. Before you beat me over the head with your expensive Teletype, let me explain myself further.

Much of the software for serial devices such as Teletypes is written in such a way that the software itself provides the parallel to serial and serial to parallel conversion and timing necessary to allow the Teletype to communicate with the computer. *[A prime example is the Motorola 6800's MIKBUG program.]* This may save the price of a UART, but it also ties down a complicated memory program to emulate a UART. I propose that instead of writing the software for a serial device, all software be written for parallel IO. If provisions are made for the "handshaking" status information, then the parallel information can be easily converted to serial information if desired by using a UART. With proper handshaking and a UART, the speed of serial IO can be made independent of the software, allowing the user to choose 110 bps or any other desired speed as presented to the UART clock inputs. All the user has to do is provide the desired clock rate. Further, such parallel handshaking would allow parallel devices such as the SwTPC TVT II to be used at speeds of several hundred characters per second instead of 10 per second.

If properly designed, a parallel IO interface can offer the following advantages:

1. External IO devices may be serial or parallel.
2. External IO devices may be made to run at a data rate up to their maximum speed.
3. External IO devices may be mixed; ie: a parallel input and serial output or vice versa.
4. Changing from one external IO device to another would require at most a change in a few jumpers.

Handshaking

The key to an effective IO scheme is a little thing we call "handshaking." Handshaking is a technique wherein two devices communicate their status to one another. For instance, if the computer gives the external IO device a signal that indicates that it has valid data available on its output port, and the external device, after accepting and processing this data then provides the computer with a signal that indicates it is ready to accept a new input, this is handshaking.

How can we provide this handshaking? Well, a UART already provides handshaking signals. For that matter, most external IO devices do or can provide the necessary signals needed for handshaking. What is needed is a way to communicate this information to the computer.

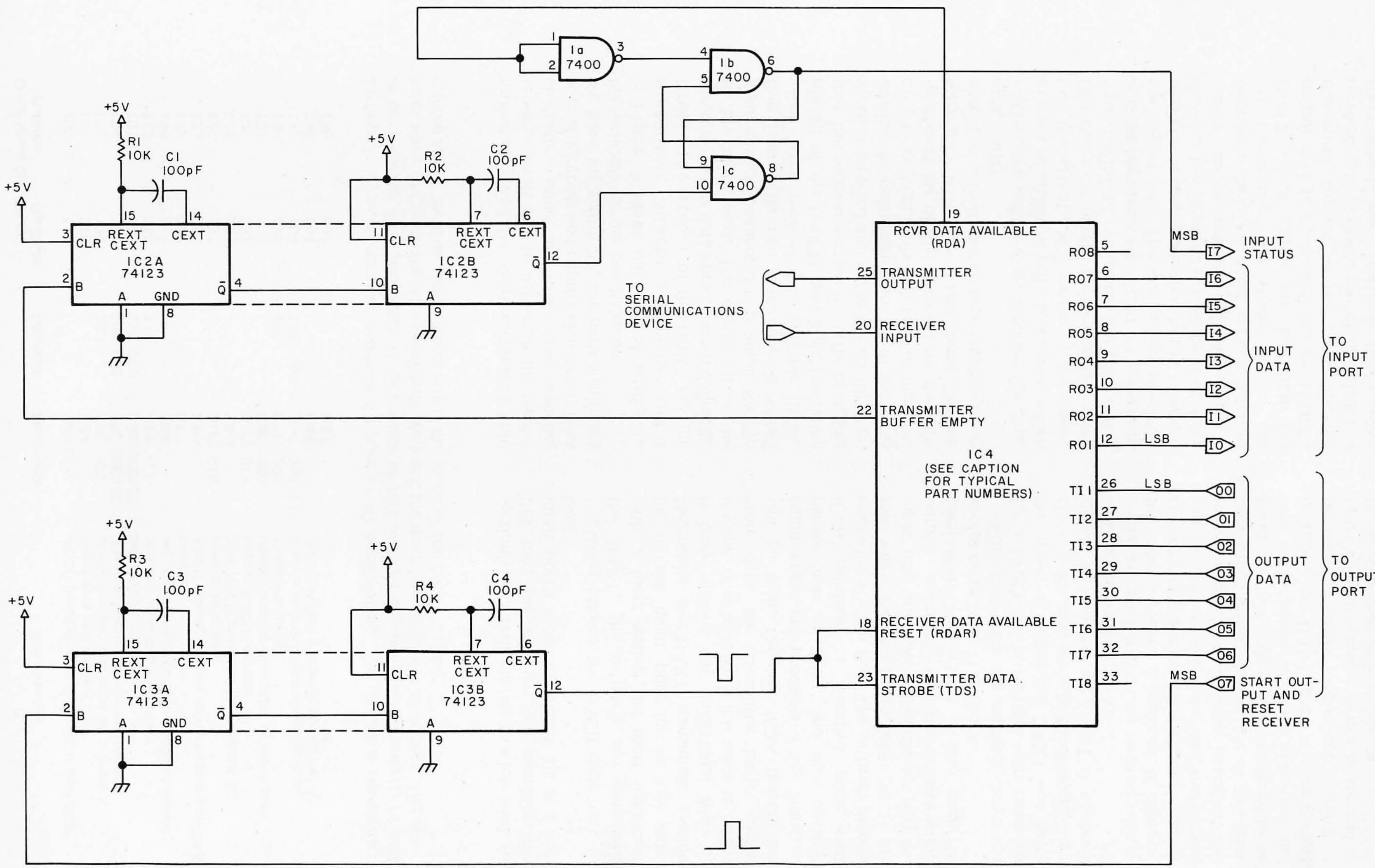
The first technique that might come to mind would be to use a special input and output port to communicate status information. As it happens, the ASCII code

→

Figure 1: A circuit for setting up handshaking between a computer and a UART. The eighth bits of the input and output ports are used to indicate whether the data has been sucessfully transmitted and the system is ready to transmit the next bit of information. This circuit can be adapted to any 8 bit computer. IC4 is a standard UART part, an asynchronous data interface. Some integrated circuit part numbers which may be used for IC4 are: the COM 2502 and COM 2017 by Standard Microsystems, the 2536 by Signetics, the AY 5 1012, by General Instruments, the TMS 6011 by Texas Instruments, the TR 1602 by Western Digital, and the S1883 by American Microsystems. All resistances are measured in ohms and all resistors are ¼ W.

Power Wiring Table

Integrated Circuit	Type	+5 V	GND	-12 V
IC1	7400	14	8	—
IC2	74123	16	8	—
IC3	74123	16	8	—
IC4	UART	1	3	2



Octal (Page 0) Address	Octal Op Code	Operand	Mnemonic	Commentary
100	113		INP 005	ASCII input routine using input port 005;
101	310		LBA	input data into register B;
102	174	200	CPE 200	check if most significant bit is high;
104	150	100 000	JTC 000100	if bit not high continue input loop;
107	106	123 000	CAL 000123	echo ASCII character;
112	044	177	NDI 177	strip off most significant bit;
114	310		LBA	registers A and B contain ASCII character;
115	007		RET	return;
123	250		XRA	ASCII output routine using output port 015;
124	133		OUT 015	clear output port;
125	301		LAB	load ASCII character into register B;
126	133		OUT 015	output ASCII character;
127	113		INP 005	check most significant bit of input port
130	074	200	CPI 200	to see if it is high;
132	100	127 000	JFC 000127	if bit is high continue checking loop;
135	301		LAB	restore ASCII character to register A;
136	007		RET	return;

Listing 1: Typical software routines for parallel IO handshaking. This listing is represented in octal for an 8008 processor. A subroutine of this sort can be used at a wide variety of data rates since the receiving device will indicate when the next byte of data is to be sent. The determining factor for data transmission rate is the speed of the UART's clock.

universally used by small computers only requires seven bits to define all the regular characters: upper and lower case, numeric, special symbols and control characters. Thus we have the eighth bit, the most significant bit, available for use as a status bit.

Figure 1 shows one way of setting up handshaking between a computer and a UART. The receiver of the UART has the seven necessary data lines connected directly to the input port. The eighth bit of the input port is connected to a simple RS flip flop formed by two cross coupled NAND gates. When receiver data available (RDA, pin 19 of the UART) goes high, indicating that the receiver section has valid data, this signal is inverted and fed to the RS flip flop, causing the most significant bit of the input port to go high. This high level at the most significant bit can be detected by the software in a wait loop. *[This status line can also be used to trigger an edge sensitive interrupt. . .*CH*]*

Once the software has acquired the data, in most situations it calls for an echo, which uses the output ASCII routine. The software first clears the output port, insuring that the most significant bit is low. Then the ASCII and the high most significant bit are sent to the output port. The low to high transition of bit O7 is detected by oneshot IC3A which causes a few hundred nanoseconds of delay before causing oneshot IC3B to output a negative pulse. This negative pulse is fed to the transmitter data strobe pin of the UART, which causes the data from the output port to be loaded into the UART's transmitter buffer. At the same time the receiver data available pin of the UART is also pulsed low, resetting the receiver section's data available line to a low level. At this time the RS flip flop maintains the input port's most significant bit at a high level.

Having echoed the ASCII code back to the UART, the software can now wait with a loop testing the most significant bit of the input port for its low state. Meanwhile the UART transmitter section is busy sending out serial data. When it is done, or at least when it is ready to accept more data, the transmitter buffer empty will go high. Oneshot IC2A detects the rising edge and produces a few hundred nanoseconds of delay, and then oneshot IC2B produces a negative pulse that resets the RS flip flop. The software detects this when the most significant bit at the input port goes low. This indicates that the output operation has been successfully completed. The system is also ready to accept more input at this time.

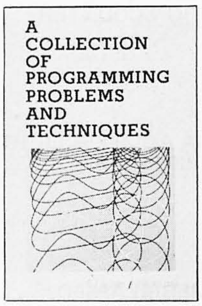
Note that while the example shown was for a UART, no restrictions were made on the speed at which the UART was being clocked. Because we used handshaking logic signals we can run our UART at whatever clock speed we desire.

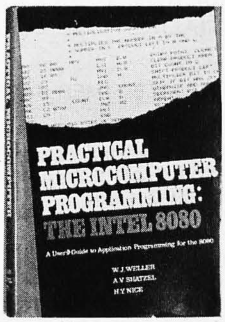
By using a handshaking method such as this, the software is reduced to periodic checking of the status bit. Output characters can be sent at any time, not overlapping receipt and echo of an input character. The only other restrictions on use of this technique are that a dummy character must be sent out when the system is initialized, and of course the UART clock rate input must be matched to the rate of the terminal or other communications device at the end of the serial transmission channel.■

We Speak Your Language...

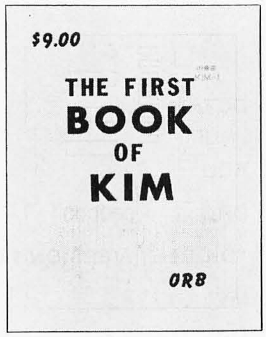
____ **Practical Microcomputer Programming: The Intel 8080** by W J Weller, A V Shatzel, and H Y Nice. Here is a comprehensive source of programming information for the present or prospective user of the 8080 microcomputer, including moving data, binary arithmetic operations, multiplication and division, use of the stack pointer, subroutines, arrays and tables, conversions, decimal arithmetic, various IO options, real time clocks and interrupt driven processes, and debugging techniques.

This 306 page hardcover book is well worth its $21.95 price and should be in every 8080 or Z-80 user's library.

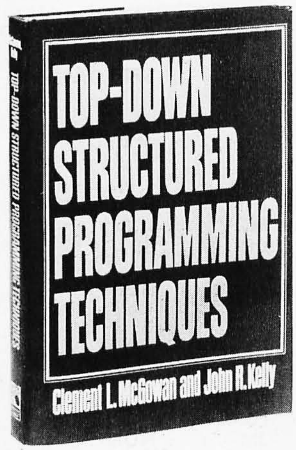




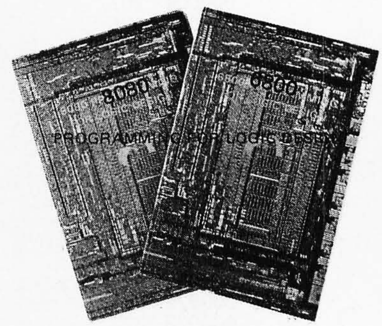
____ **A Collection of Programming Problems and Techniques** by H A Maurer and M R Williams. Here is a book that presents you with problems—nearly 400 of them: problems and games like chess, bridge, NIM; practical problems such as applications of the law of science, Kramer's rule for solving simultaneous equations, and applications of Latin squares to problems of probability; and more advanced computer science topics such as the use of Backus Naur form. One quarter of the book is devoted to an appendix that gives stymied readers hints on how to proceed with solutions of the problems. The most valuable feature of the book is its careful and thorough explanation of the use of algorithms to solve problems. No dyed-in-the-wool programmer or experimenter will be able to read this book for very long before trying to solve the tantalizing and well presented problems. $13.50.



____ **Top Down Structured Programming Techniques** by Daniel McCracken who has called structured programming "a major intellectual invention." What is structured programming? Clement McGowan and John Kelly answer this question in their lively, well written book, **Top Down Structured Programming Techniques.** Discover the three basic types of flowcharts and how to optimize them. One section deals with the best ways to manage programs being written by a team of programmers. An important feature of this book is its universality: practically any program in any language can be improved by using the ideas described in it. $15.95.



____ **The First Book of KIM.** Attention KIM users! Here is the book you've been waiting for. In it you'll find a beginner's guide to the MOS Technology KIM-1 microcomputer as well as an assortment of games including Card Dealer, Chess Clock, Horse Race, Lunar Lander and Music Box. Also featured are diagnostic and utility programs for testing both the computer and external equipment (such as cassette recorders), and chapters on expanding memory and controlling analog devices. This 176 page volume should prove an essential addition to any KIM user's library. $9.00.



____ **6800 Programming for Logic Design**

____ **8080 Programming for Logic Design** by Adam Osborne. These books are sequels to Adam Osborne's previous books, **An Introduction to Microcomputers,** volumes 1 and 2. They explain how an assembly language in a microcomputer system can replace combinatorial logic such as TTL logic and the like. If you're a logic designer, you'll discover how to do your job in a new way. If you're a programmer, you'll find many new and valuable techniques including the use of macros, high level languages, peripheral interface adapters (PIAs), and so on. Also included are complete chapters on assembler language, direct digital logic simulation, and large sections devoted to the implementation of all these ideas using the Motorola 6800 processor and the 8080 processor. $7.50 each.

DIAL YOUR ORDERS ON THE BITS **TOLL FREE HOT LINE:** 1-800-258-5477.

In New Hampshire, call collect: 924-3355

Send to:
BITS, Inc.
70 Main Street
Peterborough NH 03458

Check Payment method:
____My check is enclosed
____Bill my MC No.____________ Exp. date ______
____Bill my BAC No.____________ Exp. date ______

Name ________________________

Address ________________________

City ________ State ________ Zip Code ________

Signature ________________________

Total for all books checked $______

Postage, 50 cents per book for ______ books $______

Overseas, 75 cents per book for ______ books $______

Grand Total $______

Prices shown are subject to change without notice.
All orders must be prepaid.
In unusual cases, processing may exceed 30 days.

You may photocopy this page if you wish to leave your BYTE intact.

Circle 12 on inquiry card.

Diddle

Stan Skoglund
5757 Avenida Sanchez
San Diego CA 92124

Listing 1.

```
                                  OCTAL
                   ;
                   ;
000377             SWITCH:        EQU     255         ;ADDRESS OF SENSE SWITCH PORT
                   ;
000000                            ORG     000000
                   ;
                   ;              "DIDDLE" VERSION 1.0
                   ;
000000 076         START:         MVI     A,3         ;INITIALIZE DISPLAY PATTERN
000001 003
000002 016         BEGIN:         MVI     C,0         ;INITIALIZE DIRECTION COUNTER
000003 000
000004 107         RUN:           MOV     B,A         ;MOVE DISPLAY TO B-REG
000005 021         SPEED:         LXI     D,64        ;INITIALIZE TIMER DURATION COUNTER
000006 100
000007 000
000010 041                        LXI     H,0         ;INITIALIZE TIMER BASE
000011 000
000012 000
000013 002         DSPLY:         STAX    B           ;DISPLAY BIT PATTERN VIA ADDRESS LIGHTS
000014 002                        STAX    B           ;DO IT AGAIN TO MAKE IT BRIGHTER
000015 002                        STAX    B           ;AND BRIGHTER ....
000016 002                        STAX    B           ;AND BRIGHTER ....
000017 002                        STAX    B           ;AND BRIGHTER ....
000020 002                        STAX    B           ;AND BRIGHTER ....
000021 002                        STAX    B           ;AND BRIGHTEST!
000022 031                        DAD     D           ;ADD TO TIMER BASE.  TIMER ELAPSED?
000023 322                        JNC     DSPLY       ;NO, GO BACK AND DISPLAY AGAIN
000024 013
000025 000
000026 333                        IN      SWTCH       ;OTHERWISE READ SENSE SWITCHES
000027 377
000030 376                        CPI     0           ;DOES PLAYER WANT DISPLAY TO STOP?
000031 000
000032 312                        JZ      SPEED       ;YES, GO BACK AND RE-ISSUE SAME DISPLAY
000033 005
000034 000
000035 062                        STA     SPEED+1     ;OTHERWISE SAVE NEW SPEED
000036 006
000037 000
000040 014                        INR     C           ;BUMP DIRECTION COUNTER
000041 171                        MOV     A,C
000042 376                        CPI     63          ;TIME TO RESET DIRECTION COUNTER?
000043 077
000044 362                        JP      RESET       ;YES, JUMP TO RESET SECTION
000045 063
000046 000
000047 376                        CPI     31          ;SET STATUS WORD
000050 037
000051 170                        MOV     A,B         ;MOVE PATTERN INTO A-REG.
000052 007                        RLC                 ;SHIFT PATTERN 1 BIT TO THE LEFT
000053 372                        JM      RUN         ;JUMP IF STILL MOVING IN LEFT DIRECTION
000054 004
000055 000

000056 017                        RRC                 ;SHIFT PATTERN 1 BIT TO THE RIGHT
000057 017                        RRC                 ;SHIFT PATTERN 1 BIT TO THE RIGHT
000060 303                        JMP     RUN         ;GO DISPLAY NEW PATTERN AT NEW SPEED
000061 004
000062 000
                   ;
                   ;
                   ;
                   ;
                   ;              THIS SECTION IS EXECUTED AFTER PATTERN HAS MOVED 4 TIMES TO THE
                   ;              LEFT AND 4 TIMES TO THE RIGHT.
                   ;
000063 170         RESET:         MOV     A,B         ; MOVE PATTERN INTO A-REG.
                   ;
                   ;              THIS NEXT LOCATION CAN BE LOADED WITH OCTAL 057 WHICH WILL
                   ;              ADD A LITTLE ZIP TO DIDDLE.
                   ;
000064 000                        NOP                 ;CHANGE TO CMA
000065 303                        JMP     BEGIN       ;GO START OVER AGAIN
000066 002
000067 000
```

"Diddle" is a game program in which one can sit with an Altair 8800 computer and diddle around for some time without solving anything. The pure satisfaction of beating the game makes it all worthwhile.

The object of Diddle is to stop the moving pattern in address lights A12 and A13 while it is approaching from the right. If this is done within the rules and regulations you are considered a winner. Of course, almost everybody wins at the slower speeds, while only a selected few are talented enough to beat the computer in the high speed race. Try your luck!

Load Diddle via the front panel switches, referring to listing 1. For those of you wishing to make paper tapes or Xerox copies of this listing, be my guest. Diddle is public domain software.

Program Operation:

- Select the program starting address by setting all the address switches off (down).
- Press examine.
- Press run.
- Set address switch 10 up. Observe a pattern moving in address lights A08 thru A15. Watch it for a minute or so. Try to predict its behavior.
- Set all the address switches down. Note that the pattern stops moving. Now set address switch 10 up again. The display in the lights should start moving again if the program is operating properly.
- There exists a relationship between the speed of the moving pattern and the address switch used in the above steps. Switches to the left of A10 will cause the pattern to move faster, while switches to the right will produce a slower motion. Only A08 thru A15 can be used. Program ignores switches A00 thru A07.

Rules and Regulation:

- Once a player starts the pattern moving, he must wait at least 5 seconds before making his move to stop it. This prevents a player from creeping up to the stopping spot by toggling the speed switch.
- The moving pattern must be approaching the stopping position from the right when a player attempts his move.
- You must upon demand show that you can beat Diddle three out of three times at the speed you claim to be a winner. If you cannot do this, then you are not a bonafide winner, but just a diddler.
- In all cases the burden of proof is left to the player, not the judges. However, the judges' decision is final.

Theory of Operation and General Notes on Diddle

The display pattern is shown in the address LEDs by executing a STAX B instruction which causes the 8080 processor to output 16 bits of information to the address bus. The choice of the STAX B instruction was made so that the B register could contain the display pattern. To modify the display, it is only necessary to modify the B register and then execute the STAX B instruction. This method of display requires keeping in mind some other considerations. For instance, remember that the processor is also putting onto the address bus the location of each instruction it fetches from memory and therefore, in order to control address LEDs A8 thru A15 via the B register, the STAX B instruction must be located in low memory (below octal address 377). Try relocating Diddle to octal 10000 and you will see that the display still rotates, but address LED A12 will always be on. Also, for the same reason, the maximum display pattern one could hope to achieve would be 13 bits, although Diddle only uses an 8 bit display.

The speed of the rotating pattern is determined by adding a value to an accumulative counter until it overflows. Of course, the larger the value, the quicker the counter will overflow. The real trick with Diddle is that it lets the value be selected by the player at execution time. This is done by executing an IN instruction from port 255. MITS has provided the ability to read the front panel switches (A8 thru A15) via software using this port.

The direction of motion of the display pattern is determined by counting the number of rotations. Eight rotations will cause the pattern to revolve once, 16 rotations will cause it to revolve twice, etc. The pattern is rotated in the left direction until the counter reaches 31, then it is rotated in the right direction until it reaches 63. The counter is then reset and the cycle begins again.

It is interesting to note that the contact bounce of the front panel switches does not seem to affect the operation of Diddle. This is probably because the reflex time is much larger than the contact bounce time.■

Continued from page 69

A two-sided printed circuit board and a kit of parts are now commercially available. The board alone is $19, and the complete kit is $39. The assembled, tested and warranteed unit is $69. Send to: Meade Electronics, 511 Meade Cir, Memphis TN 38122.

address 1400, to send to the dividers, IC15 and IC19, for notes 1 and 2. Notes 2 and 3 are only in the low octaves 5 and 6 so up to three octaves may be played at once. I chose these to be in the lower octaves because I use them for rhythm and accompaniment. You may want to add other octave selectors for more control, or add manual switches to have notes 4 and 3 in any octave.

The data from the computer is strobed into the latches by write pulses generated by the address decoder, IC5 to IC7 (figure 4). $\overline{K}_5$ from KIM-1 is low whenever the sixth 1024 block of memory is addressed. K_5 is combined in IC5 with $\overline{A}_9$, $\overline{A}_8$, and R/W to produce pulses when addresses 20XX to 22XX are written into. The pulse indicating addresses 20XX (2000 to 20FF) are being written into is combined with $\overline{A}_7$ to $\overline{A}_3$ and ϕ_2 to produce a negative pulse during phase two of the computer clock when locations 1400 to 1407 are being written into. This short pulse is combined with $\overline{A}_2$ to $\overline{A}_0$ in IC7 to produce strobe pulses for the latches during the 500 ns when the data is stable on the data bus. Generating a tune is done by storing (with a timed sequence) the appropriate data into locations 1400 to 1402 as you would any other memory location. The 5 to 12 V level shifter is simply a high voltage, open collector hex inverter, 7406. In retrospect, diodes D_1 to D_4 are pro-

Nearby Musical Note		Effective Integer N	Key	Note		R Actual Ratio 256/N	Pitch Assuming 1 MHz/(16 N)	W Nearest Well Tempered Ratio	(R–W)/R
B	*	256	8	8		1.0000	244.14	1.0000	0.0000
		252	9	14		1.0159	248.02	1.0000	0.0156
C	*	242	11	11		1.0579	258.26	1.0595	0.0015
	■	240	8	15	(10, 12)	1.0667	260.42	1.0595	0.0068
		234	9	13		1.0940	267.09	1.1225	0.0260
C#	*	225	15	15		1.1378	277.78	1.1225	0.0135
		224	8	14		1.1429	279.02	1.1225	0.0178
		220	10	11		1.1636	284.09	1.1892	0.0220
D	■*	216	9	12		1.1852	289.35	1.1892	0.0034
		210	14	15		1.2190	297.62	1.1892	0.0245
		208	8	13		1.2308	300.48	1.2599	0.0237
D#	*	200	10	10		1.2800	312.50	1.2599	0.0157
		198	9	11		1.2929	315.66	1.2599	0.0255
		196	14	14		1.3061	318.88	1.3348	0.0220
		195	13	15		1.3128	320.51	1.3348	0.0170
E	■*	192	8	12		1.3333	325.52	1.3348	0.0011
F	*	182	13	14		1.4066	343.41	1.4142	0.0054
	■	180	9	10	(12, 15)	1.4222	347.22	1.4142	0.0056
		176	8	11		1.4545	355.11	1.4142	0.0277
F#	*	169	13	13		1.5148	369.82	1.4983	0.0110
		168	12	14		1.5238	372.02	1.4983	0.0167
		165	11	15		1.5515	378.79	1.5874	0.0231
G	*	162	9	9		1.5802	385.80	1.5874	0.0045
	■	160	8	10		1.6000	390.63	1.5874	0.0079
		156	12	13		1.6410	400.64	1.6818	0.0248
G#	*	154	11	14		1.6623	405.84	1.6818	0.0117
		150	10	15		1.7067	416.67	1.6818	0.0146
A	■*	144	8	9	(12, 15)	1.7778	434.03	1.7818	0.0026
		143	11	13		1.7902	437.06	1.7818	0.0047
		140	10	14		1.8286	446.43	1.7818	0.0256
A#	■*	135	9	15		1.8963	462.96	1.8876	0.0045
		132	11	12		1.9394	473.48	1.8876	0.0266
		130	10	13		1.9692	480.77	2.0000	0.0156
B	*	128	8	8	(next octave)	2.0000	488.28	2.0000	0.0000

■ Diatonic major scale notes "best fit" to A = 440 standard.
* Best fit of diatonic major scale to equally tempered scale based on B = 244.14.

Table 5: Table of possible intervals. The circuit of figures 1 to 4 produces the following set of possible frequencies assuming a 1 MHz central processor clock. In this table, outputs have been grouped near the equivalent well tempered scale ratio and frequencies. The asterisk (*) indicates best fit for a logarithmic well tempered scale series starting at a ratio of 1.0000, calculated using a program on a pocket calculator. Notations in parentheses show effective integers derived by shifting to the next octave. Note that with this calculation, use of "best" fit finds the note A in this octave at 434 Hz, 0.7% flat with respect to the standard A of 440 Hz. Table 6 picks a set from this table which is closest to the standard pitches but not optimal with respect to equal temperment.

bably not needed since IC6 probably can't supply enough current to damage IC3, even though the input voltage maximum to IC3 is specified as 5.5 V. The four extra outputs on the quad latches are used with 24 LEDs to give you a bonus light show, and are useful in figuring out what data is being sent to the interface from the computer. The LEDs are lit with 0s instead of 1s at the J inputs, so that the more lights, the lower the divisor and the higher the note. If you want the lights to read the same as the J inputs, reverse them and tie the anodes to +12 V.

The middle seven octaves of the interface each have 33 unique combinations of the key and note dividers. I've made a list of frequencies in one such octave. You'll notice right away that there is no way to get a perfect fifth if you use 244.14 Hz as the home or "tonic" note. This is because to go up in frequency by 3/2s, you need to already be dividing by a number that has 3 in it such as 9, 12, or 15. So if you want to change to a note that is a fifth from the tonic, 248.02 Hz, 260.42 Hz, 267.09 Hz etc can be used as the tonic, but 244.14 Hz, 258.26 Hz, 279.02 Hz can't be. Although this may seem restricting, remember that the octave here has almost three times as many notes as an octave on a piano. For the tritone interval(s), you will find that 10/7 and 7/5 are indistinguishable from 64/45 and 45/32, and are easier to use.

Now to get started using the interface, let's write a program to play the *do* to *do* scale in both major and minor modes. To keep it simple, we'll let note 1 play and keep the others silent. To silence notes 2 to 4, we need to store a 1 in bit 7 of location 1401 and in bits 4 and 7 of location 1402. To hear anything, we also need a 0 in bits 4 and 7 of location 1400 and to hear note 1, a 0 in bit 4 of 1401. One set of data to accomplish this is:

Address	Data
1400	xy
1401	Fz
1402	FF

where x, y, and z are numbers less than 8 and the Fs are any number more than 7 (eg: hexadecimal E).

If you look at the major and minor mode sequences in table 1, "Key of C Major," or in table 2, "Key of A Minor," you'll find that the major scale tonic must contain two 3s and a 5 in the divisor so you can multiply by 9 and 15, and the minor scale tonic must contain two 3s.

The so called "rate multipliers" also do their multiplication by dividing by a smaller number. To make this idea clear to you I've written out the ratios, divisors, output frequencies, and the data to be written into locations 1400 and 1401 (don't forget to write FF into 1402). For example, if hexadecimal 34 and hexadecimal F7 are written into 1400 and 1401, the 1.00 MHz clock will be divided by (8x12x15x2) to give an output frequency of 347.222 Hz.

To play the major scale, your memory should look like this:

Address	Data
SCALE+0	34
SCALE+1	F7
SCALE+2	40
SCALE+3	F2
etc	

The major and minor diatonic scales and the twelve note chromatic scale are not the only scales that are pleasing to the human ear. With this interface you should be able to create new and pleasing musical scales, and compose music which has never been heard before. You can explore the sounds of intervals with frequency ratios of 7/4, 9/5, 9/7, and 7/6 which are not found in Western music. You should also be able to invent some new and interesting chords since you will have more of the harmonic series available to you. Just watch out for the critical band by keeping your notes more than 50 Hz apart plus 25 Hz for each kHz.■

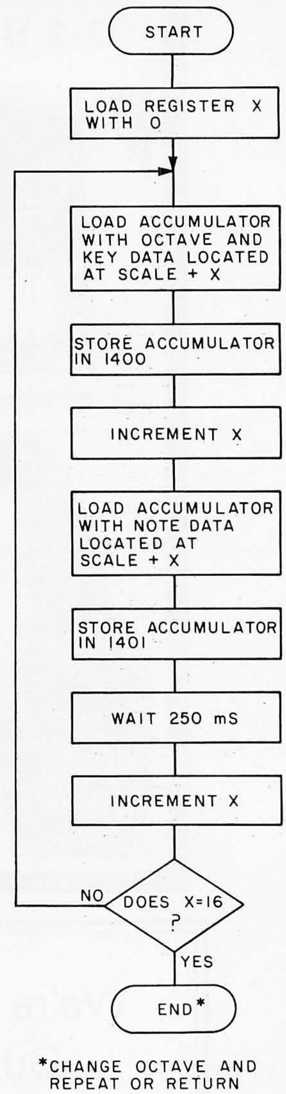


Figure 5: Flowchart of a simple program to cycle through a series of note codes found in a table indexed by register X, with 16 entries.

		Ratio	Divisor Octave	Divisor Key	Divisor Note	Data 1400	Data 1401	f_{KON1}
Major Scale	Tonic	1/1	8	12	15	34	F7	347.2
	Second	9/8	16	8	10	40	F2	390.6
	Third	5/4	8	12	12	34	F4	434.0
	Fourth	4/3	8	9	15	31	F7	463.0
	Fifth	3/2	8	12	10	34	F2	520.8
	Sixth	5/3	8	12	9	34	F1	578.7
	Seventh	15/8	8	12	8	34	F0	651.0
	Octave	2/1	4	12	15	24	F7	694.4
Minor Scale	Tonic	1/1	8	12	12	34	F4	434.0
	Second	9/8	16	8	8	40	F0	488.3
	Third	6/5	8	12	10	34	F2	520.8
	Fourth	4/3	8	12	9	34	F1	578.7
	Fifth	3/2	8	12	8	34	F0	651.0
	Sixth	8/5	4	12	15	24	F7	694.4
	Seventh	16/9	8	9	9	31	F1	771.6
	Octave	2/1	4	12	12	24	F4	868.1

Table 6: A selection of codes taken from the integers of table 5 and applied to the hardware of figures 1 to 4, to create a major scale (tonic F relative to A = 440) and minor scale (tonic A, relative to A = 440).

Continued from page 14

rescuing the small space cruiser, the captain was being questioned concerning his actions (which had necessitated the intervention of a starship). His statements were routinely routed through the data base for confirmation or contradiction. Less than one second after he stated "I have a Masters licence.", the computer countered with "Incorrect. Masters licence revoked stardate 1317.6." Even more surprisingly, the computer had previously identified this captain correctly from his voice print and other sensory data. His complete file, including criminal record, was instantly available to the *Enterprise* officers. So, at the very least, the *Enterprise* data base must contain complete biographical information on all known criminals in the galaxy![27]

The second incident began with the computer's response to the inquiry: "Computer. Linguistic banks. Definition of the following word: 'redjac'." After receiving a negative reply, all other banks were searched. The answer was then in the affirmative: "Working. Red Jack: Source: Earth, nineteenth century. Language: English. Nickname applied to mass murderer of women. Other Earth synonym: Jack the Ripper." At this point an additional inquiry was made: "Computer. Criminological files. Cases of unsolved multiple murders of women since Jack the Ripper." The computer responded with a half dozen such crimes ranging in time from the unsolved murder of seven women in Shanghai, China, Earth in 1932 to the crimes on Rigel IV only one year preceding the inquiry.[28]

The amount of linguistic, criminological and biographical data that would have to be stored on line to answer these types of questions almost defies belief.

Apart from the exceptional tasks which the *Enterprise* computers perform, their man-computer interface is even more amazing. All the computer terminals are equipped for audio IO, full graphics including the display of photographic data, and can be tied into any of the ship's numerous data banks. The voice commands that can be processed by the ship's computers are not from a small set of command words (eg: "End yellow alert! Begin processing. Find Yeoman Rand!"); rather they are completely unrestricted English! Examples of typical commands are:

Compute to the last digit the value of pi.[29] [Quite a trick if you can do it . . . CM]

Computer. Digest log recordings for past five solar minutes. Correlate hypotheses. Compare with life forms register. Ques-

tion: Could such an entity within discussed limits exist in this galaxy?[30]

Engineering Officer Scott reports warp engines damaged but repairable. Ascertain precise degree and nature of damage, compute nature and magnitude of forces responsible, and program possible countermeasures.[31]

Library computer . . . Give me everything you have on a man named or known as Kodos the Executioner. After that, a check on an actor named Anton Karidian.[32]

Hardware Innovations Necessary

With the immense amount of data that must be stored on line and available for fast access, one area of technology that must have been highly developed in the Star Trek era is memory technology. A reasonable estimate of the size of the *Enterprise* data base is 10^{22} bits. To achieve a retrieval "in a manner of seconds," an effective memory access time of 10^{-15} seconds is required (see reference on page 180). No operational memory devices today offer the necessary fast access time, along with high capacity, low power consumption and small volume. However, a number of devices hitherto restricted to the research lab are becoming operational which have some or all of the desired qualities. These include the so-called electronic disks, the Megastore of the Ampex Corporation, and Josephson junction devices.

Electronic disks are memory devices employing various new technologies which do not involve mechanical movement, but which possess the capacity of fixed or movable head disks, along with dramatically faster access times. They are just now beginning to become operational. Some problems exist in a few of the technologies, eg: memory volatility is a slight problem in the electron beam addressed electronic disk.[33,34]

The Megastore of the Ampex Corporation, however, represents a new use for an old technology. Megastore uses magnetic core, a technology that has been on the decline since the advent of smaller and faster semiconductor memories. The use of Megastore as a mass storage device is now a fact.[35]

The Josephson junction is another entirely new technology which could be used for memory. The Josephson junction device is a superconducting tunnel junction first demonstrated in 1962 by Brian Josephson at Cambridge. In one form, it exhibits a very rapid switching phenomenon between two modes of junction tunneling which enables

the device to be used as a flip flop register. The Josephson junction, as far as memory technology is concerned, is over 15 years away. One difficulty with this technology is that it operates at the extremely cold temperature of liquid helium (4.2°K) and therefore requires a sophisticated external refrigeration unit.[36] The typical access times for memory devices employing these technologies are shown in the following table:

Integrated circuits	10 to 100 ns
Magnetic core	1000 ns
Fixed head disk	8,000,000 ns
Electronic disk technologies:	
• Charge coupled devices	6400 ns
• Domain tip propagation	1,280,000 ns
• Electron-beam addressed	10,000 ns
• Magnetic bubbles	640,000 ns
Josephson junction	1 ns

Without a doubt, the new technology which provides the fastest access is the Josephson junction. (Laboratory access times of 0.6 ns have been recorded.) Because of the speed of the Josephson junction, its small size and low power consumption, much work is currently being done to bring it from the laboratory into the working computer world.

With respect to operational memory devices, then, the current technology is still five or six orders of magnitude away from the world of Star Trek, where the measure used is memory access time. However, an additional possibility is that the *Enterprise* computer may be organized as an associative or content addressable memory. Such an organization would reduce the access time requirements by at least a few orders of magnitude and would make the memory access time requirements closer to our current technology.

The remote terminals on the *Enterprise* also represent a real challenge for today's electronic technology. These terminals are primarily used for live visual communication between ships as well as for internal ship communications. They are also capable of displaying photograph quality records from a data base plus alphanumerics. Since the videophone is a reality today (although in very limited quantities and at great expense),[37] the visual communication usage alone poses no problem. Graphics display terminals using the current raster scan technology can presently provide this.

The display of photographs from the data base, however, is a different matter. If the raster technology is used, then the storage of digitized photographs by current data compression techniques[38] would require such an immense amount of secondary storage[39] that our current

memory technologies could not do the job. But one could envision a central photograph storage from which a specific photograph is retrieved mechanically and brought into position in front of a high resolution TV camera. The camera produces the necessary raster scan signal to drive the display. Whether a mechanical retrieval of a random photograph could be performed in a timely fashion, however, is open to discussion.

Another possibility is to use the plasma display technology coupled with back projection. This is currently done in the Magnavox 10000 Plasma Display Terminal, now being successfully used in the PLATO IV Project (Programmed Logic for Automatic Teaching Operation, Fourth Generation) at the University of Illinois.[40,41,42] The Model 10000 allows back projection of microfiche data concurrent with the display of graphic information.[43] Such a system allows a small amount of information on the screen to be modified without redrawing or recomputing the unchanging major portion of the display. The problem on the *Enterprise*, however, is to transmit such a back projection from a common library to a remote terminal.

Any hardware on board the *Enterprise* must be rugged. The starship is often subjected to jolts of sufficient magnitude to propel a forewarned, experienced and able-bodied crewman a distance of 20 feet or more.[44,45,46] While it is stated that the computer hardware is enclosed in antigravity, antiacceleration and antiradiation fields,[47] this is at best impossible to evaluate and at worst incomprehensible to today's science.

Software Innovations Necessary

Apart from the hardware innovations necessary to realize the level of computer technology present in Star Trek, there are software and theoretical developments which must be made. Certain elements of that level of technology are not hardware bound (or at least do not appear to be so). Two of these most apparent to the computer oriented Star Trek observer are speech recognition and semantic comprehension of a natural language.

Speech Recognition

It is often hard for nontechnical people to understand the problems involved in speech recognition by computer. After all, every 5 year old child can understand the speech of fellow human beings, even in the presence of significant background noise or degradation. Can the process be so difficult to program?

What some do not realize is that speech is a method of communication used by intelligent, knowledgeable beings who employ elements of syntax, semantics and world knowledge in order to process a complex signal. But only a portion of this signal is purely linguistic information. Current studies indicate that the 5 year old is able to utilize both hemispheres of his brain in language learning, unlike adults. Biologically, the child is specially adapted to learn language by recognizing and inferring grammar from the speech of those around him. This research brings into new light the human cognitive abilities for language and speech recognition. Recognition of the spoken word is often an underestimated task.[48,49] Star Trek presents a world in which this problem is completely solved. Almost all commands to the computer complex are given vocally. The processor is always able to convert this audio signal into an appropriate internal representation of English words and sentences.[50]

At present, the success in speech recognition could be characterized by saying that a vocabulary of 1000 words can be recognized for one speaker, or a vocabulary of one word for 1000 speakers.[51] The reasons for this lack of progress in over 20 years of work are many, but the main difficulties are presently in separating the phonemes (the basic units of speech) from each other and from other information in the speech signal.[52] Some researchers have estimated that less than 1% of the energy transmitted by voice output is used for the linguistic signal itself. The majority of the energy communicates other factors like the sex, state of health (head cold, flu, etc), emotional state and other nonlinguistic information about the speaker.[53] Merely separating the linguistic portion is difficult because it is masked by the torrent of extraneous information (as far as speech recognition is concerned).

Aside from the separation of "pure" linguistic information from the speech signal, separating the individual phonemes has also proven to be nontrivial.

Speech is a semicontinuous phenomenon. To be able to separate individual phonemes implies the difficult task of mapping this semicontinuous process onto a sequence of discrete entities. While some success has been attained with this method of phonemic analysis, current research is dealing directly with the semicontinuous phonetic stretches which correspond in some sense to fused phonemic elements such as syllables.[54]

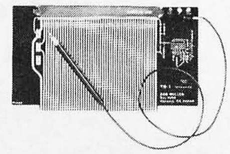
Relying merely on the audio signal alone

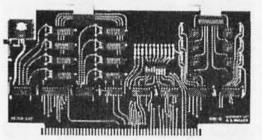
is insufficient for automatic speech recognition.[55,56] Current efforts correspond to the theories of human cognition of speech and employ elements of grammatical, contextual and semantic constraints as well as world knowledge. Techniques and models from artificial intelligence, mathematical linguistics, the theory of stochastic processes, and acoustical and phonological analysis are being extensively employed.[57,58] The difference (or more accurately the gulf) between the current state of speech recognition and that of the world of Star Trek cannot be overemphasized, however. Extensive theoretical analysis and perhaps some breakthroughs in computational techniques are required before this gulf can be bridged.

Natural Language Processing

Completely separate from the notion of speech recognition is the idea of semantic comprehension of a natural language by a computer, ie: computer "understanding" of the meaning of a normal English statement, as opposed to a statement made in a programming language, however "high" level. Suppose that some suitable input method is used (voice, if the speech recognition problem has been solved) to get English statements into the computer memory. What success has there been on the design of algorithms which could syntactically and semantically analyze these statements so that they can be "understood" by the computer? In the world of Star Trek, this problem has been completely solved. From the examples cited earlier, one can see that the *Enterprise* has an English "compiler" which can accept and correctly process unrestricted English.

The current state of natural language understanding is not yet even near that of Star Trek. However, many universities and research centers are examining this problem and have made significant contributions in the past few years. At present it is possible to correctly process relatively unrestricted English statements made in the context of narrow fields, eg: marine chemistry. While the vocabulary of these fields is understandably restricted (one rarely uses the word "sonata" while making retrievals from a chemical data base), the syntax covers a wide range of possible English constructions. One example of such a prototype language "understander" is the LUNAR system developed by Bolt Beranek and Newman Inc. This system is used to process questions to a data base of geological data recovered from lunar samples. Examples of the kind of





English that the system can correctly process are:

> *What is the average modal concentration of Illmenite in Type A rocks?*
>
> *I need all chemical analyses of lunar soil.*
>
> *Give me the K/Rb ratios for all lunar samples.*

One interesting failure of the system occurred on the following input by a nongeologist:

> *What is the average weight of all your samples?*

It seems, among other things, that the system had no notion of "ownership" built into its vocabulary, and even more important, it had no mass data on the samples anyway.[59]

Prototype work such as the LUNAR system is paralleled by (and in fact the LUNAR system employs) recent automata theory work on the nature of natural languages. While early natural language modeling employed deterministic finite state automata (with their associated regular grammars) the naivete of this model is well known. Even a pushdown automaton (equivalent to a context free grammar) is known to be an inadequate model for a natural language.[60] However, attempts to extend these models to linear bounded automata (equivalent to context sensitive grammars) or additions to the context free model have met with some objections. Recent work on transition network grammars, which are an extension of regular grammars (equivalent to finite state machines), has produced very encouraging results. In a very real sense, this model is superior to the extensions of the context free model and holds much promise for the future.[61,62] The transition network grammar model is also considered to be equivalent to the artificial intelligence approach to language comprehension used in the well-known system by Winograd.[63]

As in speech recognition, much work remains to be done before natural language comprehension by computer can reach the level of Star Trek. Most probably work in mathematical linguistics will supplement and be supplemented by practical prototype language "understanders." In the meantime, the prototype work will provide usable, albeit restricted, natural language "compilers."

It can be seen that Star Trek represents a level of computer science and technology decades beyond the current state of the art. Immense or revolutionary resources are required for the on line memory alone. Even

fitting the mainframe along with all this secondary memory into the available floor space would be difficult. Complex phenomena such as speech and language appear to be completely understood by the scientists of the Federation, yet in our own time these problems have resisted intense research efforts.

Ironically, some areas of Star Trek's computer science appear to be virtually identical to present day standards. Computer security procedures can still be undermined by ingenious programmers.[64,65] In addition the art of computer programming seems to be not much advanced from today. At one point the Science Officer requires a matter of hours to initiate a class 1 priority command.[66] One would think that in two centuries of software development, such urgent requests could be more easily processed and resources reallocated without extensive action by systems programmers.

We should not scoff at the view of computer science presented by Star Trek, however. Science fiction often predicts future science fact with remarkable accuracy. Lunar exploration and the laser are but two examples of current technology predicted by science fiction writers. While one can present various and often contradictory explanations of this phenomenon, one cannot dismiss it. Perhaps Star Trek does give us a glimpse of the future of the computer; perhaps not. In any case, one can agree with the *Enterprise's* Science Officer when he says, "There is one thing we can say: they will have many interesting adventures."[67]

Computer Space Aboard the Enterprise

According to the *Enterprise* blueprints,[68] the computers aboard the *Enterprise* are physically located in four areas. The primary hull contains the main ship's computer and the main engineering computer. The main ship's computer is located on both decks 7 and 8 below the bridge, with a total of about 1400 square meters of floor space. The main engineering computer is located in the engineering spaces on the seventh deck below the starboard energy converters. This complex has about 64 square meters of area.

The secondary hull houses the auxiliary ship's computer beneath the botany and hydroponics labs on deck 19. The total area of this complex is 340 square meters. The auxiliary engineering computer, also housed in the secondary hull, is located on deck 16 near the shuttlecraft hanger. This complex occupies about 72 square meters.

The total space aboard the *Enterprise* devoted directly to the computer complexes is therefore about 1900 square meters. This figure does not take into account the numerous terminals located throughout the ship.

Memory Requirements for the Enterprise

The *Enterprise's* computer contains "the whole of human and humanoid knowledge"[6] in the 23rd century and "in a matter of a few seconds . . . can obtain an answer to any factual question".[9] More specific information on the volume of data to be stored is required in order to gauge the memory requirements, though. By making reasonable and plausable assumptions, this amount of information can be expressed in bits in order to estimate the memory size and access time requirements needed aboard the *Enterprise*.

At present, the Library of Congress contains over 72 million volumes.[69] It would seem reasonable to assume that the store of knowledge in the *Enterprise* is about a million times that of the present Library of Congress (given the present rate of growth of technical knowledge and assuming that this rate will increase steadily over the next 250 years). This yields a bank of knowledge for the *Enterprise* which would fill about 10^{14} volumes. Assuming a volume contains on the average 1000 pages of 1000 words, with each word averaging between six to eight letters, and a 6 to 8 bit character code for computer storage, the information expressed in those 10^{14} volumes represents a staggering amount of bits: on the order of 10^{22}!

It is much more difficult to arrive at an estimate of the access time needed to meet the stated user interaction. The entire notion of record retrieval on secondary keys is not nearly as well formulated as primary key retrieval.[70] (Recall that retrieval by a primary key is a retrieval of *one* record based on the field which orders the data set. A secondary key retrieval is a retrieval of *all* records which have a given value, or a given range of values, in any other field. An example of a primary key retrieval is: "Retrieve the medical record for Pete Smith." An example of a retrieval on a secondary key is: "Retrieve *all* cases of cancer diagnosis in the past year."

Knowing the current state of memory technology, though, one can linearly extrapolate from response times on data bases of known size in use today to get an approximation of the memory access times needed to achieve the desired response for the data base on the *Enterprise*. Knuth presents an example of a fully inverted file (a file together with a dictionary/directory for each field) with one million records of 40 characters each. With some reasonable assumptions, he computes an access time of 10.7 seconds for a request involving ten fields for which ten records are found. The average access time for the disk used was 71 ms.[71] Extrapolating this example, an average memory access time of 10^{-15} seconds would be required to process this retrieval in one second if the data base was as large as that on the *Enterprise* (assuming a similar data organization). But current work in retrieving entries from large data bases may significantly shorten the required memory access times.[72]

FOOTNOTES

1 Asimov, Isaac, "The Last Question," Science Fiction Quarterly, November 1956, page 6.

2 Heinlein, Robert A, The Moon is a Harsh Mistress, Putnam, New York, 1966.

3 Clarke, Arthur C, 2001 — A Space Odyssey, New American Library, New York, 1968.

4 Jones, D F, Colossus (aka: "The Forbin Project"), Putnam, New York, 1967.

[5]Gerrold, David, **When Harlie Was One**, Doubleday, New York, 1972.

[6]"Wolf in the Fold," Star Trek episode, first telecast 12/22/67, adapted by James Blish, **Star Trek 8**, Bantam, New York, 1972, page 127.

[7]**Ibid**, page 127.

[8]In the unauthorized transmission of certain portions of the Star Fleet Technical Manual which took place on stardate 3113, the index of the manual listed these sections. The sections themselves, unfortunately, were not transmitted. Compiled by Franz Joseph, **Star Fleet Technical Manual**, Ballantine Books, New York, 1975, page 00.00:05 and pages 00:00:08 to 00:00:09.

[9]In "Space Seed," a Star Trek episode about suspended animation space ships ("sleeper ships"), we are told that Kahn, the leader of those on the sleeper ship, left Earth after 1996. Spock informs us that sleeper ships were last used in 2018. Kahn "slept" for two centuries or more, which implies that this episode takes place in the late 22nd or 23rd century. "Space Seed," Star Trek episode, first telecast 2/16/67, adapted by James Blish, **Star Trek 2**, Bantam, New York, 1968, page 106.

[10]Whitfield, Stephen E and Roddenberry, Gene, **The Making of Star Trek**, Ballantine, New York, 1968, pages 128, 40, 170 and 198.

[11]Star Fleet Headquarters, an artifically constructed "world," is the largest, "Articles of the Federation," chapter 7, article 52, paragraph 3 and article 53, paragraph 1; compiled by Franz Joseph, **Star Fleet Technical Manual**, Ballantine Books, New York, 1975, page 00:01:09.

[12]**Ibid**, Preamble, page 00:01:00.

[13]Moore, J E (Ed), **Jane's Fighting Ships 1975-1976**, pages 428 to 429, and **1976-1977**, page 565.

[14]"Articles of the Federation," chapter 7, article 53, paragraph 1: **op cit**, page 00:01:09.

[15]"Star Fleet Technical Order 03:11:20," compiled by Franz Joseph, **Star Fleet Technical Manual**, Ballantine Books, New York, 1975, page 03:11:20.

[16]"Mirror, Mirror," Star Trek episode, first telecast 10/6/67, adapted by James Blish, **Star Trek 3**, Bantam, New York, 1969, page 75.

[17]"Star Fleet Technical Order 01:04:10," compiled by Franz Joseph, **Star Fleet Technical Manual**, Ballantine Books, New York, 1975, page 01:04:10.

[18]"Outboard Bow and Stern Elevations Blueprints," drawn by Franz Joseph, **BOOKLET OF GENERAL PLANS: USS Constitution Class**, Ballantine Books, New York, 1975, sheet 4.

[19]Cogswell, T R and Spano, C A, **Spock, Messiah!**, Bantam, New York, 1976, page 5.

[20]"Tomorrow is Yesterday," Star Trek episode, first telecast 1/26/67, adapted by James Blish, **Star Trek 2**, Bantam, New York, 1968, page 27.

[21]"Journey to Babel," Star Trek episode, first telecast 11/17/67, adapted by James Blish, **Star Trek 4**, Bantam, New York, 1971, page 51.

[22]"Balance of Terror," Star Trek episode, first telecast 12/15/66, adapted by James Blish, **Star Trek 1**, Bantam, New York, 1967, page 74.

[23]"For the World is Hollow and I Have Touched the Sky," Star Trek episode, first telecast 11/8/68.

[24]"The Lights of Zetar," Star Trek episode, first telecast 1/31/69, adapted by James Blish, **Star Trek 6**, Bantam, New York, 1972, page 26.

[25]"Return to Tomorrow," Star Trek episode, first telecast 2/9/68, adapted by James Blish, **Star Trek 9**, Bantam, New York, 1973, page 3.

[26]"The Enterprise Incident," Star Trek episode, first telecast 9/27/68, adapted by James Blish, **Star Trek 4**, Bantam, New York, 1971, page 27.

[27]"Mudd's Women," Star Trek episode, first telecast 10/13/66.

[28]"Wolf in the Fold," **op cit**, pages 132 and 134.

[29]**Ibid**, page 139.

[30]**Ibid**, page 133.

[31]"Tomorrow is Yesterday," **op cit**, page 32.

[32]"The Conscience of the King," Star Trek episode, first telecast 12/8/66, adapted by James Blish, **The Star Trek Reader II**, Dutton, New York, 1977, pages 121 to 122.

[33]Martin, R R, "Electronic Disks in the 1980s," **Computer**, volume 8, number 2, February 1975, page 24.

[34]Kelly, J, "The Development of an Experimental Electron Beam Addressed Memory Module," **Computer**, volume 8, number 2, February 1975, page 32.

[35]Kinnucan, P, "A New Use for Core Memory," **Mini-Micro Systems**, volume 10, number 3, March 1977, pages 23 to 26.

[36]Miller, R T, "Superconducting Tunnel Junctions," **IBM Research Reports**, volume 9, number 1, 1973.

[37]Hardeman, L J,"Picturephone to Change its Image," **Electronics**, volume 46, number 19, 13 September 1973, pages 75 to 76.

[38]McCarthy, J P, "Automatic File Compression," **Proceedings of the International Computing Symposium 1973**, Davos, Switzerland, North Holland, Amsterdam, 1974, pages 511 to 516.

[39]An image with a resolution approximately equal to that of a present day television frame contains about 10^6 bits. High quality, high resolution photographs can contain as many as 10^{10} bits; Ware, W H, "The Ultimate Computer," **IEEE Spectrum**, volume 9, number 3, March 1972, pages 84 to 91.

[40]National Science Foundation **MOSAIC**, volume 3, number 3, Summer 1972.

[41]Alpert, D and Bitzer, D L, "Advances In Computer Based Education," **Science**, volume 167, 20 March 1970, pages 1582 to 1590.

[42]Bitzer, D L and Johnson, R L, "PLATO: A Computer Based System Used In the Engineering of Education," **IEEE Proceedings**, volume 59, number 6, June 1971, pages 960 to 968.

[43]Model 10000 Plasma Display Terminal Brochure, MAGNAVOX, Fort Wayne IN.

[44]"Tomorrow is Yesterday," **op cit**, page 26.

[45]"The Alternative Factor," Star Trek episode, first telecast 3/30/67, adapted by James Blish, **Star Trek 10**, Bantam, New York, 1974, page 1.

[46]"The Immunity Syndrome," Star Trek episode, first telecast 1/19/68, adapted by James Blish, **Star Trek 9**, Bantam, New York, 1973, pages 179 and 183.

[47]"Outboard Bow and Stern Elevations Blueprints," **op cit.**

[48]Lenneberg, Eric H, "Capacity for Language Acquisition," Lester (Ed), **Readings in Applied Transformational Linguistics**, second edition, Holt Rinehart, New York, 1973.

[49]Lenneberg, Eric H, "On Explaining Language," Lester (Ed), **Readings in Applied Transformational Linguistics**, second edition, Holt Rinehart, New York, 1973.

[50]For the sake of simplicity, it will be assumed that all speech recognition and natural language comprehension is for only one language: English. This is clearly not the case, but it is representative. If all the problems with these two types of processing were solved for English, then the extension to other languages would be easy. **[A debatable point . . . CM]**

[51]Flanagan, J L, **Speech Analysis, Synthesis, and Perception**, Springer-Verlag, New York, 1972, page 196.

[52]Garvin, P L, "A Linguist's View of Language-data Processing," P L Garvin (Ed), **On Machine Translation**, McGraw-Hill, New York, 1963, page 20.

[53]**Ibid**, page 18.

[54]**Ibid**, pages 19 to 20.

[55]Newell, A, et al, **Speech Understanding Systems: Final Report of a Study Group**, North-Holland, New York, 1973.

[56]"The C-MU Speech Recognition Project," **Proc IEEE Systems Sciences and Cybernetics Conference**, Pittsburgh PA, 1970.

[57]Woods, W A, "Motivation and Overview of BBN Speechlis: An Experimental Prototype for Speech Understanding Research," **IEEE Symposium on Speech Recognition**, Carnegie-Mellon University, 1974.

[58]Baker, J K, "The Dragon System — An Overview," **IEEE Symposium on Speech Recognition**, Carnegie-Mellon University, 1974.

[59]Woods, W A, "Progress in Natural Language Understanding — An application to lunar geology," **AFIPS Conference Proceedings**, volume 42, 1973,

pages 441 to 450.

[60]Chomsky, N, "A Transformational Approach to Syntax," J A Fodor and J J Katz (Ed), **The Structure of Language**. Prentice-Hall, Englewood Cliffs NJ, 1964.

[61]Woods, W A, "Transition Network Grammars for Natural Language Analysis," **Comm of the ACM**, volume 13, number 10, October 1970, pages 591 to 606.

[62]Winograd, T, **Understanding Natural Language**, Academic Press, New York, page 43.

[63]Wilks, Y, "Parsing English I," Charniak, E and Wilks, Y (Ed), **Computational Semantics**, North-Holland, Amsterdam, 1976, page 99.

[64]In "Court Martial," the Enterprise Records Officer is able to modify the ship's computer based log, a file which should never be alterable; "Court Martial," Star Trek episode, first telecast 2/2/67.

[65]In "The Menagerie," the Enterprise Science Officer is able to tie in the ship's life-support system with the helm. The ship's computer would not accept any course changes, since this would adversely affect the life support. In this way, the Science Officer was able to take over the ship; "The Menagerie," Star Trek episode, first telecast 11/17/66 and 11/24/66 (a 2 part episode).

[66]"Wolf in the Fold," op cit, page 138.

[67]"Assignment: Earth," Star Trek episode, first telecast 3/29/68.

[68]**BOOKLET OF GENERAL PLANS: USS Constitution Class**, drawn by Franz Joseph, Ballantine Books, New York, 1975.

[69]Golenpaul, A (Ed), **Information Please Almanac**, 31st edition, Dan Golenpaul Associates, 1977, page 295.

[70]Knuth, D E, **The Art of Computer Programming**, volume 3 (Sorting and Searching), Addison-Wesley, Reading MA, 1973, pages 550 to 551.

[71]Ibid, pages 553 to 554.

[72]Hardgrave, W T, "The Prospects for Large Capacity Set Support Systems imbedded within Generalized Data Management Systems," **Proceeding of the International Computing Symposium 1973**, Davos, Switzerland, North-Holland, Amsterdam, 1974, pages 549 to 556. ■

Continued from page 8

types which can be declared, as well as file and record structures missing from BASIC. PASCAL is a block structured language allowing multiple character strings for procedure and data names, and is thus closer to the natural symbolic thought processes of designing a program than is BASIC.

A classical contrast between the two languages in this area of features is to pose the problem: How would I use the language to include complex numbers for use in engineering analysis or physics? In BASIC, I might not even want to consider the possiblity of using the language for complex numbers because of the kluge that would result. Using PASCAL, I would simply use the type extensibility of data to declare a complex number type and code various procedures to implement complex number operations. An example of this concept, which involves no features not inherent in PASCAL's definition, is given on pages 42ff of the *PASCAL User Manual and Report* quoted earlier. Of course, perhaps not all possible or desirable features were included in PASCAL's definition, so dialects may occur there as well as in BASIC. But the necessity of dialects generated through extensions is probably less in PASCAL, making the standard created by Professor Wirth a closer approximation to what actually gets implemented.

Lots of Implementations of BASIC Are Available.

Here is where BASIC no doubt has a considerable lead over PASCAL at the present time. But PASCAL is rapidly gaining in a catch up mode. As noted earlier, there are presently nearly 100 different implementations of PASCAL, mostly for minicomputers and larger computers ranging in size and scope up to a CRAY-1 implementation of PASCAL. At the low end, according to the *PASCAL Users Group Newsletter*, number 8, page 64, there are presently compilers implemented for the Motorola 6800, Intel 8080 and Zilog Z-80 microprocessor architectures (although the listing did not mention whether the compilers were self-compilers or cross compilers). Implementations are coming, part of the history of the language and the active following it has among computer science people.

Much Personal Use Applications Software Already Exists in BASIC.

No argument here. The number of books and periodicals which publish programs in BASIC will probably exceed the number with PASCAL representations of equivalent programs for a long time to come. But this is equivalent to saying that BASIC has been around longer in the public eye, for given time much of the same sort of software can and will be written in PASCAL as more and more implementations become available.

BASIC is Friendly.

BASIC is fundamentally an interactive approach to programming in which pro-

U.S. POSTAL SERVICE

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION

(Required by 39 U.S.C. 3685)

1. TITLE OF PUBLICATION: BYTE The Small Systems Journal
A. PUBLICATION NO.: 1 0 2 4 1 0
2. DATE OF FILING: 9/29/77
3. FREQUENCY OF ISSUE: MONTHLY
A. NO. OF ISSUES PUBLISHED ANNUALLY: 12
B. ANNUAL SUBSCRIPTION PRICE: $12.00
4. LOCATION OF KNOWN OFFICE OF PUBLICATION *(Street, City, County, State and ZIP Code) (Not printers)*: 70 Main St. Peterborough, Hillsboro, NH 03458
5. LOCATION OF THE HEADQUARTERS OR GENERAL BUSINESS OFFICES OF THE PUBLISHERS *(Not printers)*: 70 Main St. Peterborough, Hillsboro, NH 03458
6. NAMES AND COMPLETE ADDRESSES OF PUBLISHER, EDITOR, AND MANAGING EDITOR

PUBLISHER *(Name and Address)*: Virginia Peschke 74½ Grove St. Peterborough, NH 03458

EDITOR *(Name and Address)*: Carl Helmers Noone Ave. Peterborough, NH 03458

MANAGING EDITOR *(Name and Address)*: None

7. OWNER *(If owned by a corporation, its name and address must be stated and also immediately thereunder the names and addresses of stockholders owning or holding 1 percent or more of total amount of stock. If not owned by a corporation, the names and addresses of the individual owners must be given. If owned by a partnership or other unincorporated firm, its name and address, as well as that of each individual must be given.)*

NAME	ADDRESS
BYTE Publications Inc.	70 Main St. Peterborough, NH
Virginia Peschke	74½ Grove St. Peterborough, NH
Carl Helmers	Noone Ave. Peterborough, NH

8. KNOWN BONDHOLDERS, MORTGAGEES, AND OTHER SECURITY HOLDERS OWNING OR HOLDING 1 PERCENT OR MORE OF TOTAL AMOUNT OF BONDS, MORTGAGES OR OTHER SECURITIES *(If there are none, so state)*

NAME	ADDRESS
NONE	

9. FOR COMPLETION BY NONPROFIT ORGANIZATIONS AUTHORIZED TO MAIL AT SPECIAL RATES *(Section 132.122, PSM)*
The purpose, function, and nonprofit status of this organization and the exempt status for Federal income tax purposes *(Check one)*

☐ HAVE NOT CHANGED DURING PRECEDING 12 MONTHS ☐ HAVE CHANGED DURING PRECEDING 12 MONTHS *(If changed, publisher must submit explanation of change with this statement.)*

10. EXTENT AND NATURE OF CIRCULATION	AVERAGE NO. COPIES EACH ISSUE DURING PRECEDING 12 MONTHS	ACTUAL NO. COPIES OF SINGLE ISSUE PUBLISHED NEAREST TO FILING DATE
A. TOTAL NO. COPIES PRINTED *(Net Press Run)*	90,928	107,400
B. PAID CIRCULATION 1. SALES THROUGH DEALERS AND CARRIERS, STREET VENDORS AND COUNTER SALES	19,128	21,196
2. MAIL SUBSCRIPTIONS	67,665	77,651
C. TOTAL PAID CIRCULATION *(Sum of 10B1 and 10B2)*	86,793	98,847
D. FREE DISTRIBUTION BY MAIL, CARRIER OR OTHER MEANS SAMPLES, COMPLIMENTARY, AND OTHER FREE COPIES	559	1,760
E. TOTAL DISTRIBUTION *(Sum of C and D)*	87,352	100,607
F. COPIES NOT DISTRIBUTED 1. OFFICE USE, LEFT OVER, UNACCOUNTED, SPOILED AFTER PRINTING	2,275	6,793
2. RETURNS FROM NEWS AGENTS	1,301	none as of this da
G. TOTAL *(Sum of E, F1 and 2—should equal net press run shown in A)*	90,928	107,400

11. I certify that the statements made by me above are correct and complete. SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER: V. A. Peschke Publisher

12. FOR COMPLETION BY PUBLISHERS MAILING AT THE REGULAR RATES *(Section 132.121, Postal Service Manual)*

39 U. S. C. 3626 provides in pertinent part: "No person who would have been entitled to mail matter under former section 4359 of this title shall mail such matter at the rates provided under this subsection unless he files annually with the Postal Service a written request for permission to mail matter at such rates."

In accordance with the provisions of this statute, I hereby request permission to mail the publication named in Item 1 at the phased postage rates presently authorized by 39 U. S. C. 3626.

SIGNATURE AND TITLE OF EDITOR, PUBLISHER, BUSINESS MANAGER, OR OWNER

PS Form Mar. 1977 3526 *(Page 1)* (See instructions on reverse)

grams are entered in source form and tested within the confines of one session with effectively instant change from editing to execution. If PASCAL is to become an equivalent "friendly" language, it must be implemented in a way which allows a similar instant change from editing the design to trying out the design of an application.

Whether this friendliness requirement can be best met by an interpreter or a compiler is an open question, but it is a definite requirement. In BASIC the rule to date has been interpretive, or semicompiled code, where semicompiled means that symbols for language tokens are replaced by compact codes. In PASCAL to date, compilation has been the rule rather than the exception. It is conceivable that a compiled PASCAL coupled with an editing and object code maintenance facility oriented to the block level might give sufficiently quick response at the terminal with much faster execution times associated with compiled code.

Another open question concerning PASCAL is that of how much memory is required for a PASCAL self-compiler or resident interpreter in a typical personal computer's microprocessor based system. I suspect that a compiler or interpreter of PASCAL can be built which will fit within 16 K to 32 K bytes of memory, but whether this is really possible or not is by no means clear to me.

To sum up the thesis, PASCAL is well on its way to becoming the kind of widely known language which will be taught as a matter of course to new students of programming. This in turn will tend to boost the long term acceptance of PASCAL and get it established as one of the major languages, a process which at an earlier date occurred for FORTRAN and BASIC. For our own part, we at BYTE are interested in giving PASCAL a boost. We have a survey article about PASCAL in preparation at the present time. We would also like to talk to implementors of the language who would be interested in marketing PASCAL compilers or interpreters through software book publications which include source code and machine readable object code. For those who desire more background information on PASCAL, we recommend the PASCAL User's Group, run by Andy Mickel at the University of Minnesota Computer Center, 227 Exp Engr, University of Minnesota, Minneapolis MN 55455, (612) 376-7290. The *PASCAL Newsletter* is published four times per year, and at the time of this writing costs $4 per year.■

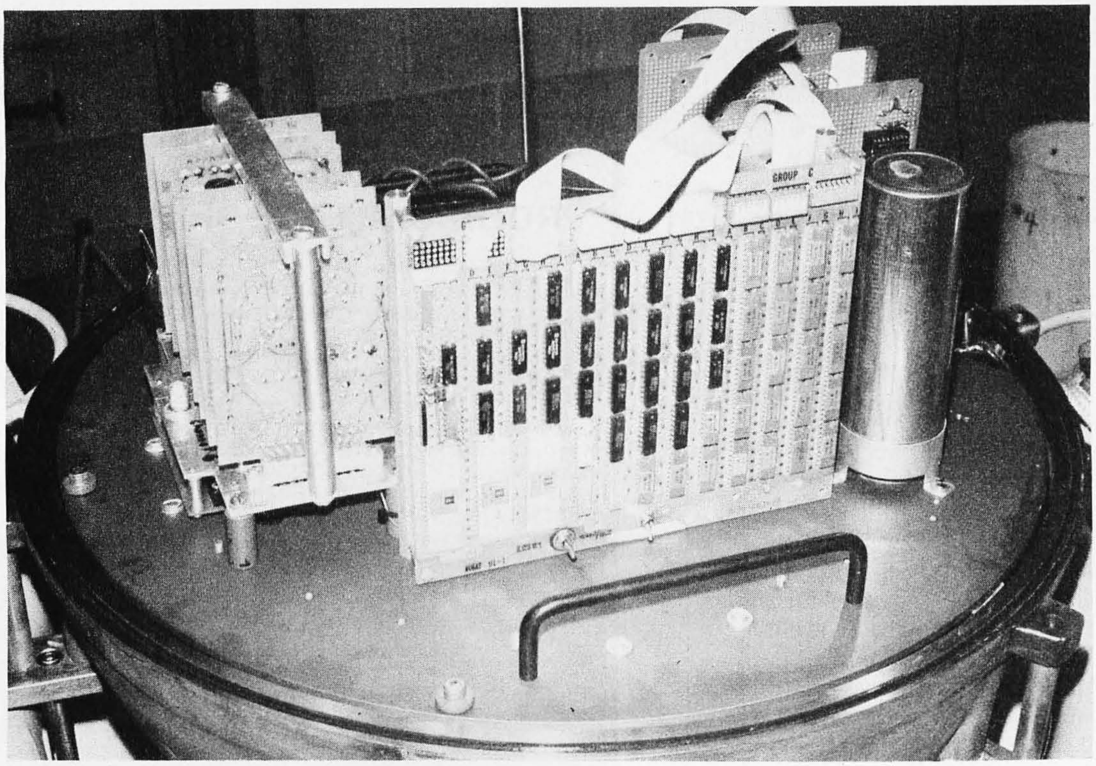


Photo 1: The Intercept Jr unit shown in the titanium sphere with the cover removed. The rebuilt processor board is the one closest to the front of the assembly. The system is used to collect data from the continental shelf.

Henry Lahore
Oceanography WB-10
U of Washington
Seattle WA 98195

A User's Report on the Intercept Jr

In August 1976 Intersil Inc brought out Intercept Jr, a battery operated microcomputer system using Intersil's IM-6100 12 bit CMOS microprocessor, which has been described in the May and June 1976 issues of BYTE and the June 1976 IEEE Proceedings.

At the time, we in the oceanography department at the University of Washington needed a microcomputer for underwater data acquisition. This required a battery operated system which would have to operate for several months. We had to choose between the Intercept Jr or the RCA COSMAC processor, the only battery powered systems available at the time. Because the Intersil processor recognizes the instruction set of the Digital Equipment Corporation PDP-8E minicomputer, and because we had cross assemblers and programmers available on campus who were familiar with the PDP-8 language, we chose the Intersil product.

The Intercept Jr microcomputer system consists of a 10 by 11 inch (25.4 by 27.9 cm) board which features a keyboard, two 4 digit LED displays, a monitor with microinterpreter in read only memory, 256 words of programmable memory, three printed circuit card sockets for option cards, and a battery power supply for $281. Option cards available include 1 K bytes of programmable memory, 2 K bytes of bipolar programmable read only memory, a UART, and an audiovisual display card using LEDs and a speaker. Excellent hardware and software documentation is included.

We have made three different battery powered data acquisition systems using Intercept Jr. The sediment motion system to be described has analog and digital sensors and uses the IM-6100 microprocessor with 2000 words of memory. Data is recorded on a Memodyne incremental low power tape cassette.

In our sediment motion data acquisition system (see photo 1) we have removed the microprocessor from its socket on Intercept Jr and put it in our data acquisi-

About the Author

Henry Lahore obtained a masters degree in electrical engineering at the University of Washington in 1970, and has worked with Raytheon, Computer Automation, and Prime Computers in the areas of hardware interfacing and programming. This project is his first experience with CMOS circuitry.

tion system, which consists of an Augat wire wrap board with 2000 words of programmable memory, two CMOS type parallel interface elements, and miscellaneous circuitry. Two ribbon cables with DIP connectors form an umbilical cable of 28 conductors between the processor and the data acquisition system.

We use a cross assembler for PDP-8E assembly language on a DEC PDP-10 computer on campus. The 2000 word program is loaded over the phone lines into our system in about 3 minutes at 300 bps. The Intercept Jr system monitor accepts the standard BIN loader format outputted by PDP-8 minicomputers or cross assemblers. At the end of program loading, a checksum is displayed on the numeric display to show if there were any errors in transmission. We get perfect transmission more than 90% of the time.

While in the lab, the data acquisition system uses Intercept Jr to print memory dumps automatically at the end of an averaging time. This has been extremely useful for debugging the program. When we are satisfied with its operation, the Intercept Jr is removed from the data acquisition system. The latter is then sealed in a titanium sphere, and the entire apparatus is brought out to the continental shelf to record data for up to eight weeks. Thus, in this project, the Intercept Jr serves as a removable peripheral device used for program debugging, loading and testing.

When the data comes back on tape cassette, we use the Intercept Jr system to dump the data to the Teleprinter for a quick look, and then dump the data over the phone lines to the PDP-10 computer for data analysis.

The data acquisition program is designed to operate in read only memory with 256 words of programmable memory. After loading the program over the phone lines, we throw a switch which disables write commands to all but the lower 256 words of the memory. This switch, which we label RAM/ROM, has been very useful. We use the Intersil IM-6561 256 by 4 bit CMOS programmable memory. Harris Semiconductor is coming out with a field programmable CMOS read only memory, HM-6612, which should be an exact replacement for the IM-6561 programmable memory. This will eliminate having to load the program every time, but will increase power consumption. Each read only memory will consume 10 mA while being addressed.

Power Consumption

For any electronics which is to run on batteries for an extended period of time, power consumption must be in the low milliwatt range. Many microprocessors consume more than 500 mW by themselves. Our CMOS data acquisition system consumes 18 mW of power. The CMOS circuit family has very low power consumption and features almost as many different types of circuits as the TTL family.

The Intercept Jr CMOS microprocessor system comes with four alkaline D cells that last for about 20 hours of operation with the LED numeric display on. The following current measurements are for operation with the 2.45 MHz crystal oscillator installed:

	5 V	6 V
HALTED	4.4 mA	5.6 mA
RUN	23 mA	30 mA
RUN with display	223 mA	350 mA

We see that the LED display takes ten times as much power as the rest of the Intercept Jr.

Power consumption in the CMOS computer system is proportional to the number of instructions per second executed. Our data acquisition system samples data once per second, does a little computation, and then has nothing to do until the start of the next second. Following Intersil's application bulletin, M005, we turn off the system clock while waiting for the next second. This reduces power consumption during the pause by a factor of 5.

The IM-6100 has a WAIT line that allows the use of slow memory. This WAIT state can be used to reduce power consumption for all of the system except for the microprocessor itself. Turning off the clock to the IM-6100 reduces current from 10 mA to .25 mA at the standard clock frequency.

We have also slowed the system clock to 1 MHz. This allows us to increase the size of the pullup resistors from 1 k to 20 k ohms in order to reduce power consumption. Our data acquisition system has to wait while various peripherals are ready; slowing the system clock reduces the number of times we have to execute a wait loop. The reduced clock rate also permits us to use long cables between the Intercept Jr and our data acquisition system. The instantaneous current consumed by our data acquisition system is 10 mA. The average consumption is only 3 mA due to its being on only about 1/3 of every second.

Are you in the process of designing a turn-key computer system?

Do you have problems interfacing the hardware?

Then you need the hardware specialist
COMPUTER MACHINE SERVICE.

CMS is in the business of assembling and repairing of microcomputers and equipment.

CMS is not in the business to sell computer equipment.

Our primary objective is to provide assembly service of the highest quality as well as prime quality repair service on microcomputers and data communications equipment.

For information call: **213-328-9740**
or **213-328-9760**

Or write: **Computer Machine Service**
2909 Oregon Court
Torrance, CA 90503

FOR ALL YOUR HARDWARE PROBLEMS CMS IS THE ONE TO CALL.

Circle 31 on inquiry card.

BYTESHOP™
the affordable computer store

7825 BIRD ROAD
(305) 264-2983
DIAL 264-BYTE

OF MIAMI

WE HELP YOU GET YOUR SYSTEM UP AND RUNNING.

IMSAI 8080	MEMORY EXPANSION
BYTE-8	COLOR TV GRAPHICS
SWTP MP68	LEAR SIEGLER ADM 3
CROMEMCO	PAPER TAPE READER

PROCESSOR TECH
INTERFACES (KITS or ASSEMBLED UNITS)

IN FORT LAUDERDALE

1044 E Oakland Pk Blvd

(305) 561-2983

Circle 14 on inquiry card.

ELCOM ENTERPRISES
Now Distributing....

HAZELTINE 1500 - Basic Terminal [kit or assembled]

CONTROL DATA - Conversational Display Terminals [assembled]

ADDS - Regent 100 and 200 Terminals [assembled]

BEE HIVE'S - B100 and B500 [assembled]

TWX - New and Used

TELEX - New and Used

SHUGART - Mini-floppy Disc Drive Units $310.00 while they last!!!

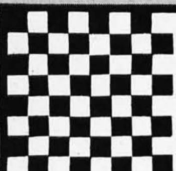
ALL PRICES FOB CONCORD, TN.
BankAmericard/Visa and Mastercharge welcome

ElCom ENTERPRISES
10728 DUTCHTOWN RD.
CONCORD, TN. 37720
(615) 966-3352
(615) 689-5559 - Call 24 hours.

B-12-77

DEALERS INQUIRES INVITED

Circle 47 on inquiry card.



CHESS BASIC

Chess Basic is a chess program that runs on extended Basic. It has a length of 7K characters and plays an intermediate level game. Send for a sample run to see how well it plays.

CHESS BASIC $10.00

add $1.00 for shipping
California residents add 6%

T K ELECTRONICS
PO BOX 1832
Fremont Calif. 94538

Circle 120 on inquiry card.

TYSON'S CORNER, VIRGINIA
(WASHINGTON, D.C. AREA)

THE COMPUTER SYSTEMS STORE
"Specializing in Systems"

FEATURING:

Processor Technology	Lear Siegler
Poly Morphic	DEC
Diablo	SWTPC
E & L	Seals
Vector Graphics	Digital Group
Apple	TDL
CROMEMCO	Books — Magazines

COMPUTERS FOR HOME, SCHOOL, & SMALL BUSINESS: Our staff will help you select from the best of each manufacturer to complete the system best suited to your needs.

SPECIAL: LEAR-SIEGLER ADM 3A

CSS COMPUTER SYSTEMS STORE

1984 CHAIN BRIDGE ROAD, McLEAN, VA. 22101
TELEPHONE (703) 821-8333

OPENING SOON:
ALEXANDRIA, VA • NORFOLK, VA • BALTIMORE, MD • ORLANDO, FLA

Circle 33 on inquiry card.

THE PROM SETTER
READ/WRITE
1702A and 2708
CAN ALSO WRITE and READ 2716
AND OTHER EPROMS

ALTAIR/IMSAI COMPATIBLE
NO EXTERNAL POWER SUPPLY

LET YOUR COMPUTER DO IT ALL

SOFTWARE INCLUDED
Doubles as an 8 Bit Parallel I/O

KIT COMPLETE—$210
ASSEMBLED—$375

DELIVERY LESS THAN 60 DAYS

SZERLIP ENTERPRISES
1414 W. 259th STREET
HARBOR CITY, CA. 90710
Calif. Res. Add 6% Sales Tax

Circle 113 on inquiry card.

SUPPLIES



- FLOPPY DISKS, MINI OR STANDARD MEMOREX OR 3M
- 3M DATA CARTRIDGES DC300A, DC100A
- 3M DIGITAL CASSETTES
- 3M OR MEMOREX AUDIO CASSETTES, C-60
- 3M DISK CARTRIDGES

WE OFFER:

- COMPETITIVE PRICING
- IMMEDIATE DELIVERIES (Any Quantity)
- UNCONDITIONAL GUARANTEE

BETA BUSINESS SYSTEMS
8369 VICKERS ST., #G
SAN DIEGO, CA 92111
(714) 565-4505

Circle 11 on inquiry card.

REMEMBER
16K
ASSEMBLED AND TESTED
$520
Features;

STATIC OPERATION
ACCESS TIME 250ns
S100 BUS COMPATIBLE
LOW POWER TYPICALLY
205ma 8v
105ma 16v
24ma 16v neg
BATTERY BACKUP

AVATAR
AVATAR SYSTEMS

3830 SWENSON
BOX 223
LAS VEGAS, NEV. 89109
(702) 733 8304

Circle 134 on inquiry card.

IT'S A GREAT BIG COMPUTER WORLD
But You Only Need

THE COMPUTER CORNER

- SOL – A New Dawn Is Here!
- IMSAI 8080
- POLY – 88
- TDL Z-80
- Memories & I/O Boards

- Computer Book Service
- Magnetic Tapes & Disks
- Full Line of Magazines
- Brain Games & Puzzles
- Workshops & Club Information

Visit THE COMPUTER CORNER for all your computer needs. Stop in and browse – you'll like our personal service.

THE COMPUTER CORNER
White Plains Mall – Upper Level
200 Hamilton Avenue
White Plains, New York 10601

Tel: (914) WHY - DATA

Ample Parking
10-6 Daily & Saturday
10-9 Thursday

Circle 23 on inquiry card.

High Speed Operation

We have not had any need for high speed operation, but the Intercept Jr system can be made to run at fairly high speeds by increasing the voltage supply. The typical instruction takes 10 minor cycles for execution:

Voltage (V)	OSC Frequency (MHz)	Execution Speed (cycles per second)
5 V	2.54 MHz	200,000
6 V	4.0 MHz	350,000
10 V	8.0 MHz	700,000

The 8 MHz operation requires the use of the "A" version of the microprocessor. This operates faster than the PDP-8E minicomputer but takes less than .01 the power.

Option Cards

The UART card required the most modification for our use. It has been designed for operation only at 110 bps. We added the Fairchild CMOS 4702 clock generator to allow selection of rates up to 2400 bps. The early UART cards were produced with RS-232 connectors mounted on the wrong side of the board. We also had to add a negative voltage source. Our Texas Instruments teleprinter requires that a jumper be put on the UART card between pins 4 and 8. We found it very useful to add an LED to indicate when data was being received over the phone lines.

The 1 K bytes of CMOS programmable memory that is available from Intersil has two AA cells mounted on the card for battery backup. This card draws only 10 μA from the AA cells when power is turned off. It is affected by static electricity if it is removed from its socket. Touching the printed circuit side of the removal board causes a latchup on the card, scrambling the data saved in memory.

The AUDVIS (audiovisual) board (6957) has 12 slide switches for inputs, and four LED numeric displays with 12 LEDS in parallel for visual output, in addition to a speaker for audio output.

We intend to introduce the microprocessors and the data acquisition system to scientists and technicians with the AUDVIS board, because it allows the new user to quickly get used to microcomputer input and output, and binary to octal conversion. It was developed for use in 1 day Intercept Jr classes given by Intersil around the country.

CMOS Logic Probe

The Intercept Jr system is meant to be only a tutorial system. If one gets around to doing much hardware development, a CMOS logic probe becomes almost a necessity. For operation with the 2.45 MHz crystal, the response time of the probe must be faster than 300 ns. Continental Specialties, among others, has a fast CMOS logic probe with a useful pulse latch feature that turns on an LED if there has been a logic state change. The hobbyist might be interested in the AVR electronics probe which we have also used.

Selecting a Low Power Microcomputer

As a brief postscript we note that, as of this writing, the available battery powered processors on the market include the Intersil IM-6100, RCA COSMAC, Texas Instruments 990 I^2L 16 bit microprocessor, and an upcoming I^2L version of the NOVA minicomputer from Fairchild. We found it difficult to obtain a cross assembler for any low powered computer other than the Intersil CMOS PDP-8E, however.■

·····INTRODUCING NEMESIS 7·····

THE GAME WITH THE BRAIN

The first of a series of handheld computerized games using "State of the Art" technology



Nemesis7 electronically duplicates exactly the ancient game of Nim. Pebbles were arranged into 3 columns of 7, 5, & 3 pebbles each. Play alternated with an opponent. Each could remove any or all pebbles from a column but could manipulate only one column per move. The loser took the last pebble.

Your respectable opponent in Nemesis7 is a fixed program computer consisting of a custom designed LSI computer chip or circuit. To win is extremely difficult but possible. Human opponents are also permitted.

Easy to assemble kit includes all parts (except battery) including custom MOS computer circuit, PC board & case. Some soldering required........................ $ 9.95

Kit that requires no soldering.................. $14.95
Complete factory assembled game................. $16.95
Battery extra (9V transistor radio)............. $ 1.50

Include $1.50 shipping   Nevada residents add 3½ PST
Send check or money order to:

Computer Creations Inc.
118 West 2nd Street; Reno, Nevada; 89501

Circle 24 on inquiry card.

WASH. DC AREA HOBBYISTS!

Micro Diversions

announces

HOBBYIST MICROCOMPUTING

An introduction to the design, construction, and programming of microprocessor-based computers.

EIGHT WEDNESDAY EVENING SESSIONS
JANUARY 4 - FEBRUARY 22

These are companion "how-to-do-it" courses on the hardware and software technologies of personal computing taught by Micro Diversions hardware and software designers. Call/send/circle for brochure.

HARDWARE - $75.00 SOFTWARE - $75.00
BOTH $125.00

Classes limited in size; early registration is recommended.

P.O. BOX 527
FALLS CHURCH, VA. 22046
(703) 533-1133

Circle 72 on inquiry card.

South Florida

SUNNY COMPUTER STORES, INC.

South Florida's First Computer Store

WE CARRY:

- ALPHA MICRO SYSTEMS, MICRO FILE, IMSAI
- BOOKS, MAGAZINES, NEWSPAPERS
- FULL SERVICE
- SOFTWARE
- WRITE OR PHONE US FOR PRICES

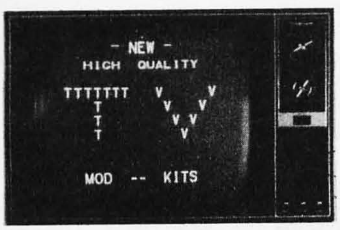
MORE THAN JUST TOYS

Monday 12 noon – 9 pm
Tues.–Fri. 10–6

UNIVERSITY SHOPPING CENTER
1238A SOUTH DIXIE HIGHWAY
CORAL GABLES, FLA. 33146
(305) 661-6042

Circle 110 on inquiry card.

CONVERT ANY TV
TO A HIGH QUALITY MONITOR



- **Hot Chassis or Transformer sets**
- **64-80 characters per line**
- **By-passes tuner & I.F.**
- **Normal viewing unaffected**
- **Safe—Easy installation**

ACVM Hi-Resolution $24.95 ppd

RFVM Ch2-6 Modulator $9.95 ppd

VAMP INC. Box 29315
Los Angeles, Calif. 90029

Calif. Residents add 6% Sales Tax

Circle 125 on inquiry card.

TELETYPE
RECONDITIONING AND REPAIR

Prices for reconditioning of complete machinery including parts:

MODEL NO.	PRICES
M-32 ASR,KSR	$250.00
M-33 ASR,KSR	$250.00
M-35 ASR,KSR	$295.00
M-30 RO	$195.00
M-33 RO	$195.00
M-35 RO	$195.00

ALL PRICES FOB CONCORD, TN.

BankAmericard/Visa and Mastercharge welcome

The prices listed above are for mechanically complete teletype machines only. Quotes for repair of incomplete machines made on request.

EI ElCom ENTERPRISES
10728 DUTCHTOWN RD.
CONCORD, TN. 37922
(615) 966-3352
(615) 689-5559 - Call 24 hours.

Circle 47 on inquiry card.

WHAT IS a really good way to protect all your valuable back issues of BYTE that will keep them handy, clean, and organized?

Send to: BYTE, POB 5120, Philadelphia PA 19141
for information on binders and files.

MAKE YOUR COMPUTER PRODUCTIVE WITH SOFTWARE WRITTEN IN NORTHSTAR BASIC

REM General Ledger and Statements $75
REM Mailing Labels $50
REM Federal Income Tax $200, Schedule A and B
And More to Come . . .
REM System Descriptions $2 each

REM System supplied on Diskette with Program Documentation and Operating Manual

RUN Benson & Costello
29-02 23 Avenue
Astoria, New York 11105
212-932-3316

Circle 10 on inquiry card.

computer depot® Inc.
(Credit Cards Accepted)
3515 W. 70th St., Mpls., MN 55435
(612) 927-5601

- ☐ Motorola DII 6800 Evaluation Kit. $235.00
- ☐ National SCMP Kit with Keyboard $190.00
- ☐ E&L Digi-Designer Kit—Design Your Own Circuits. . . . . $ 77.75
- ☐ Speech Lab Kit—Talk To Your Computer $249.00
- ☐ 2708 Eproms Full Spec 450NS From TI & Fairchild . . . . $ 16.95
- ☐ IMSAI 8080 Complete 22 Slots All IC's with Sockets—Assem. & Tested . . . $850.00
- ☐ TTL Catalog and Remember, We Pay The Shipping.

Name ____________________
Address ____________________
Part No. ____ Qty. ____ Amt. Enclosed ____

Circle 26 on inquiry card.

CANADIANS!

Introducing our kit-by-the-month plan available for only $500.00 down and $150.00 per month.

(Write for more information)

IMSAI 8080   KIT: $897.50
ASSM: $1245.20

Canadian Duty and Federal Tax Included

Hobby systems from $999.00 (Kit).

Business/engineering systems from $11,900.00. (Assembled and Installed*).

Educational discounts available. We will develop custom application packages. Contact us for further information. Send $1.50 for catalogue.

VISA • CHARGEX ACCEPTED

Rotundra Cybernetics

Box 1448, Calgary, Alta. T2P 2H9
Phone (403) 283-8076

(Installation outside Western Canada extra).

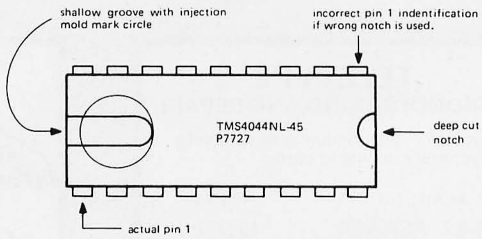
Circle 97 on inquiry card.

A Note to Novice Kit Builders . . .

Be certain that when you are putting together a kit by soldering in your integrated circuits that the identity of pin 1 position is unambiguous. A particular example we recently learned about by personal experience was the Texas Instruments TMS-4044NL 4 K memory part used by a novice to assemble an 8 K memory board from a kit product. The instructions for this particular kit quite properly stated that:

> *. . .the "dot" or "notch" on the end of the package is used for orientation purposes and must match with that shown on the component layout drawing for each of the integrated circuits.*

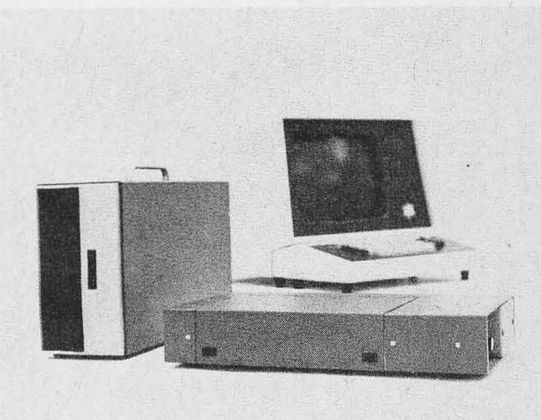
The only problem was that the part (for the novice) was ambiguous:



The first-time novice who assembled the board in question chose the deep notch as the identification of the proper end of the IC for pin 1, and was 180° from the proper orientation. The only other confirmation of proper orientation would be that pin 1 is at the left when the markings on the IC are read in the normal fashion. Later inspection of the mechanical data section of a TI manual confirmed that the proper orientation of the package in this case was given by the shallow rectangular groove.■

What's New?

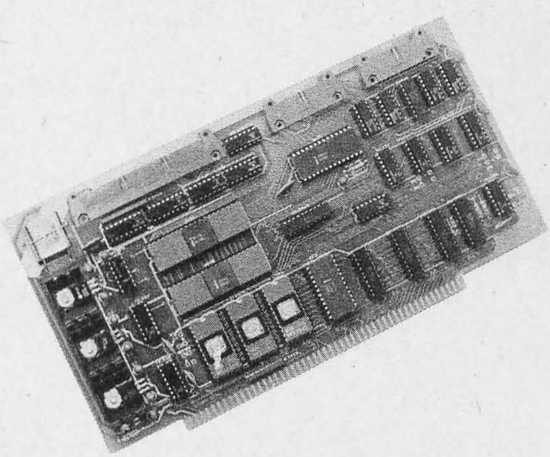
Program Development System



This new portable program development system is manufactured by Zilog, 10460 Bubb Rd, Cupertino CA 95014. At $2850 in single unit quantities, the user gets a system which includes a 300 K byte full size floppy disk, 16 K bytes of user programmable memory, 3 K bytes of internal programmable read only memory, and an RS-232 or current loop interface to a terminal. Add a terminal to this box, and individual users may find the price quite attractive as well as the commercial and industrial users for whom the system was intended.■

Circle 579 on inquiry card.

8085 Processor Card



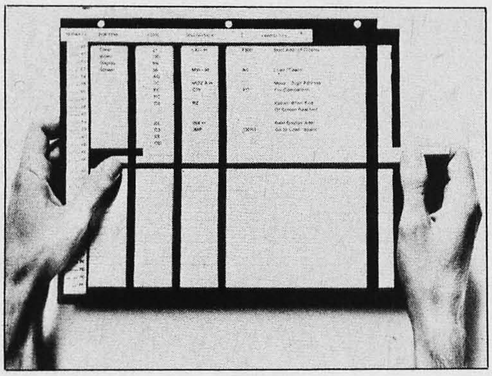
Space Byte, 1720 Pontius Av, Suite 201, Los Angeles CA 90025, makes this $499 8085 processor card for the S-100 bus. The board includes a monitor program, RS-232 serial ports and interface for the iCOM 3700 or Frugal Floppy disk systems.■

Circle 580 on inquiry card.

Circle 21 on inquiry card.

What's New?

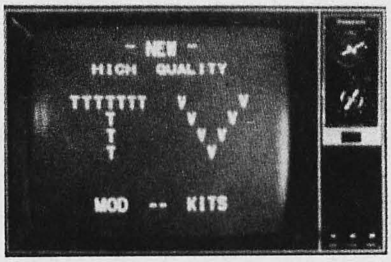
New Aid for Machine Language Hackers



Microcomputing enthusiasts who lack an assembler and must program in machine language (without the benefit of a power eraser) may appreciate this new product. The Basic Operational Programming Aid (BOPA) comes with 32 removable, reversible plastic slats on which machine code can be written, one byte to a slat. Machine instructions can be inserted or rearranged simply by moving the slats around. A set of BOPAs can be carried in a 3 ring binder. The basic BOPA is $11.95, with a special ink pen for $1 and a solvent solution for $1.25. An expanded 256 entry system is also available for $74.95, from Vamp Inc, POB 29315, Los Angeles CA 90029.■

Circle 521 on inquiry card.

Two Ways to Convert Your TV to a Monitor

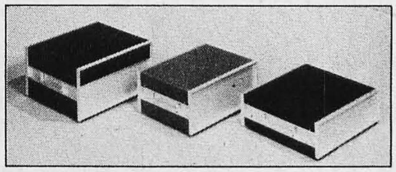


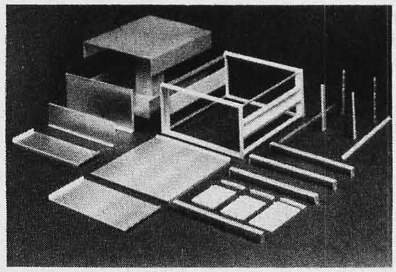
There are two ways to convert your TV set to a video monitor: You can use the video signal to modulate a high frequency (RF) signal, creating a tiny transmitter which is attached to the antenna terminals of your TV set, or you can physically modify the set to enter the video signal just beyond the TV's video detector. The former method requires no modifications to the TV, but limits the bandwidth of the signal and hence the number of dots making up characters across the face of the screen.

You can take your choice of these methods with two kits from Vamp Inc. The RFVM-1 kit ($8.95) includes a 1.25 by 1.75 inch (3.2 by 4.4 cm) printed circuit board which is designed to be installed inside your video source and attached to the TV through the antenna terminals, yielding a frequency response of up to 4 MHz. The ACVM-1 kit ($23.95) includes a 2 by 3 inch (5.1 by 7.6 cm) baord designed to be installed inside the TV set and provides a frequency response up to 10 MHz. A bypass switch is included with the ACVM-1 to permit normal TV program viewing. Both kits are complete with all parts and can be obtained with a $1 shipping and handling charge from Vamp Inc, POB 29315, Hollywood CA 90029.■

Circle 523 on inquiry card.

Attention Homebrewers: Roll Your Own Case with This Package

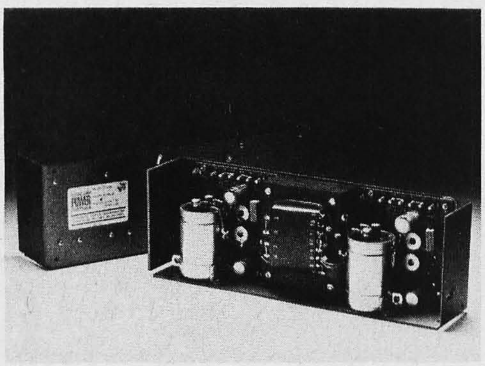




The "Flex-i-pak," a flexible instrument housing system, is supplied in a basic unit frame configuration with a variety of chassis and brackets to choose from as needed. Extrusions, brackets and panels contain a pattern of holes on ½ inch centers, allowing easy "erector set" construction. Card guides may be installed in a variety of ways. The basic case, measuring 17 inches (43.1 cm) by 13, 16 or 20 inches (33.0, 40.6 or 50.8 cm) with heights from 3.5 to 12.25 inches (8.9 to 31.1 cm), features vinyl covered top and bottom, side rails and perimeter frame of extruded aluminum. Price of the Flex-i-pak in single quantities varies from $72 to $145 from the Buckeye Stamping Company, 555 Marion Rd, Columbus OH 43207, (614) 445-8433.■

Circle 524 on inquiry card.

New Power Supplies



Circle 522 on inquiry card.

The new SPS-D and SPS-T multiple output DC power supplies are designed for microprocessors and memories, floppy disks, and similar applications. Available with adjustable voltages from ± 5 VDC to ± 28 VDC and currents from 1 to 12 A, the units feature complete isolation between outputs, current limiting, short circuit protection, and optional overvoltage protection. Prices range from $104 to $139 and weight from 7 to 16 pounds. Additional technical information and a new catalog are available from Standard Power Inc, 1400 S Village Way, Santa Ana CA 92705, (714) 558-8512.■

Attention Readers, and Vendors. . .

Where Do New Product Items Come From?

The information printed in the new products pages of BYTE is obtained from "new product" or "press release" copy sent by the promoters of new products. If in our judgment the neat new whizbang gizmo or save the world software package is of interest to the personal computing experimenters and homebrewers who read BYTE, we print the information in some form. We openly solicit such information from manufacturers and suppliers to this marketplace. The information is printed more or less as a first in first out queue, subject to occasional priority modifications.

Bit Status Display

Greg Tomalesky
164 Preston Rd
Parsippany NJ 07054

Here is an interesting modification to the 7 segment display circuit shown in figure 3 of "LEDs Light Up Your Logic" in February 1976 BYTE, page 54. The original circuit displayed 1 or 0 to indicate the corresponding logic state. With the addition of two more 7437 inverters and some rewiring (see figure 1), the circuit will display H and L in place of 1 and 0. I used a DL-704 common cathode display, but of course a common anode display may also be used.

Carrying this idea a little further, you can display HI and LO in a similar manner by using two displays.

7 segment displays can be used to display many other letters such as A, C, E, F, H, I, J, L, O, P, S and U in upper case, and b, c, d, h, i and o in lower case. The upper case letters A, C, E, F and the lower case b and d may be used in conjunction with the numerals 0 thru 9 to form hexadecimal displays for use with microcomputers such as the M6800 and MC6502, which have hexadecimal based structures.■

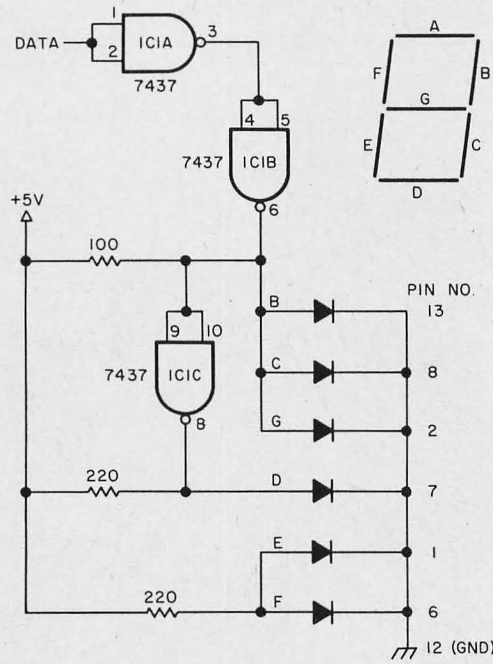
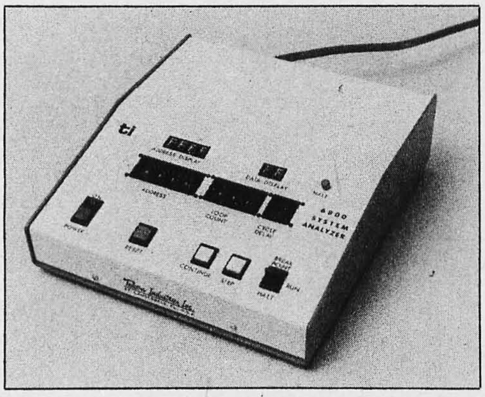


Figure 1: This 7 segment display will monitor the state of a single bit and output an H or L depending on the status of the data input. Power connections to the 7437 inverter are +5 V to pin 14 and ground to pin 7. All resistances are measured in ohms and all resistors are 0.25 W.

What's New?

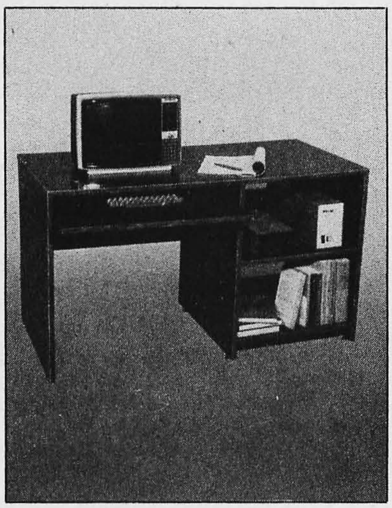
6800 System Analyzer



Designed for use with the Motorola 6800 microprocessor, this system analyzer features single step instruction execution, a hardware implemented breakpoint which shifts from full speed to single step mode when a selected address is reached, loop counting, and cycle delays. Hexadecimal addresses, loop counts and delays are set up with thumbwheel switches and displayed on LEDs. Priced at $995, the unit is offered by Telcon Industries Inc, 5701 NW 31st Av, Fort Lauderdale FL 33309, (305) 971-2250.■

Circle 525 on inquiry card.

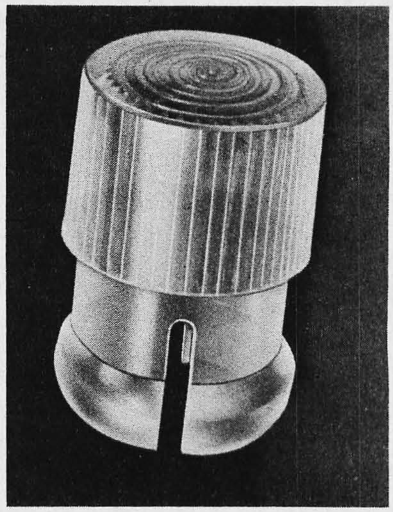
Furniture for the Micro Age



A desk specifically designed to house a personal or business microcomputer system is now available from Computer Systems Design. The "Microdesk" (a trademark) can be easily assembled without tools in five minutes. The desk is constructed of high density vinyl clad board and measures 48 by 24 by 28 inch (122 by 61 by 71 cm). A sliding shelf at convenient typing height is provided for a keyboard. Also included for equipment and books are two fixed shelves, one of which is adjustable. Available as options are additional shelves or rails for rack mounted equipment. The Microdesk is available for $96.50, FOB Wichita, from Computer Systems Design, 1611 E Central, Wichita KS 67214, (316) 265-1120.■

Circle 527 on inquiry card.

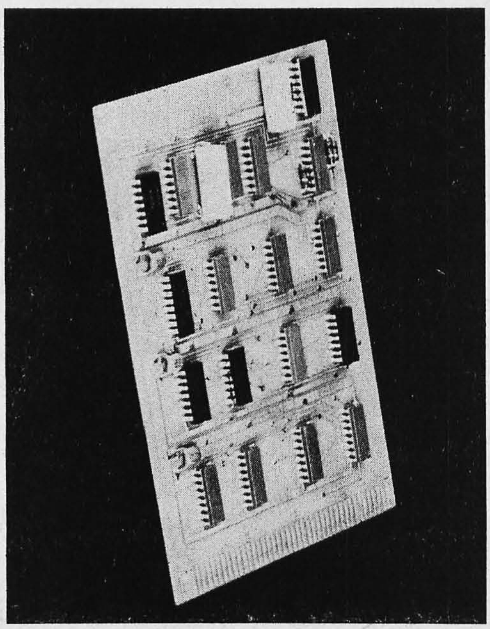
A Quick Way to Panel Mount LEDs



This one-piece lens and panel mount for T 1¾ light emitting diodes, called Cliplite, snaps securely into a ¼ inch hole and requires no tools to assemble. A point source LED used with the Cliplite produces a display five times brighter than a diffused LED without any loss in viewing angle. Five colors: red, green, amber, yellow and clear allow the Cliplite to enhance and provide uniformity of color output from LED to LED. The units are priced at $.10 in 10,000 quantities. A free sample and data sheet can be requested on a company letterhead from Visual Communications Company, POB 986, El Segundo CA 90245, (213) 822-4727.■

Circle 528 on inquiry card.

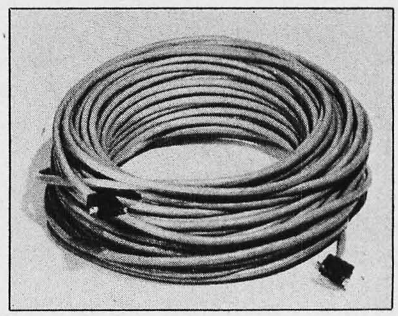
Hard Copy from Digital Panel Meters



This adapter converts digital panel meter outputs in binary coded decimal (BCD) to Teletype inputs, for hard copy and paper tape records of panel readings. The Model DPT-415 has a present format that enables printing of up to five digits, a sign and an additional character. It generates spaces, carriage returns and line feeds. The DPT-415 is complete and runs on a 5 V power supply. The driving circuitry for the 20 mA Teletype connection is included on the 6.5 by 4.5 inch (16.5 by 11.4 cm) printed circuit board. The unit sells for $275, or $95 in quantities of 100, from Digital Laboratories, 600 Pleasant St, Watertown MA 02172, (617) 924-1680.■

Circle 526 on inquiry card.

If It's a Long Way from Your Terminal to Your Computer. . .



A new line of RS232C EIA data cables is available in lengths up to 250 feet and longer. The cables feature low capacitance (12 to 14 pF per foot), twisted wires to reduce crosstalk and shielding to protect against electrostatic noise. The cable sells for $18.50 for two RS232C end connectors plus $.75 per foot, from Data Set Cable Company, 7 Danbury Rd, POB 622, Ridgefield CT 06877, (203) 438-9023.■

Circle 529 on inquiry card.

Circle 9 on inquiry card.

All Prime Quality — New Parts Only

Lowest Prices — Satisfaction Guaranteed

EDGE CARD CONNECTORS:

Bifurcated Contacts. Not tin. *Gold over nickel.* 50/100 Pin (.100, 1.25, + .156 Pin spacing). Double Read out.

50/100 Altair Type — Dip Solder Pins	$4.25 ea	5 pcs	$4.00 ea
50/100 Imsai Type — Dip Solder Pins	$4.25 ea	5 pcs	$4.00 ea
50/100 Imsai Type with Guides	$4.50 ea	5 pcs	$4.25 ea
Imsai Guides only	$ .30/pr	5 pr	$ .25/pr

All other contacts available: Solder Eyelet, Wire Wrap, etc. *All gold.*
Many other types available: 10/20, 15/30, 18/36, etc.
When ordering: *specify type of contact.*

25 PIN SUBMINIATURE CONNECTORS:

Gold Plated Contacts.

DB25P Plug	$2.90 ea	5 pcs	$2.75 ea
DB25S Socket	$3.90 ea	5 pcs	$3.75 ea
DB51212-1 Hood (grey)	$1.00 ea	5 pcs	$ .95 ea
DB51226-1A Hood (black)	$1.00 ea	5 pcs	$ .95 ea

SAVE:
Buy a complete set:
1 Plug, 1 Socket, 1 Hood (any) $7.00/set
5 sets $6.50/set

2708 1KX8 PROM 450 NS
$18.00 ea
5 pcs or more $17.00 ea

8080A — PRIME
$13.00 ea
5 pcs $12.00 ea

Write for larger quantity discounts.

Dealers welcome.

Minimum order $10.00: Add $1.00 for shipping and handling. Orders over $25.00: We pay the shipping: Calif. residents add 6% tax. No COD's: For immediate shipment send money order or cashier's check.

When ordering: State method of shipment: Mail or UPS.

ORDER FROM:

Beckian Enterprises

P.O. Box 3089 Simi, Calif. 93063

ESAT-100
Economical Stand Alone Terminal
Assembled, Tested, Burned-In........$239.00

For Teletype, SCMP, Jolt, Kim SBC 80 Users: A reliable, inexpensive, self-contained Communicating Terminal, completely assembled, burned-in, and tested...$239.00

SPECS: 32 character x 16 lines x 2 pages of 5 x 7 dot matrix 64 character ASCII communicating with a serial, asynchronous 11 unit code, TTL compatible from 300 to 9600 baud. Keyboard Controls are back and forward space, line feed, clear page, or to end of line, select page 1 or 2, full/half duplex, local/remote, cursor on/off, odd/even/no parity. Output to TV Monitor is Composite Video, 75 Ohms. Keyboard required is parallel output 7 unit ASCII with negative true strobes. Keyboard may tap up to 200 Mo. from the ESAT-100 on-board 5V power supply. Power required is 110VAC @ 7 watts.

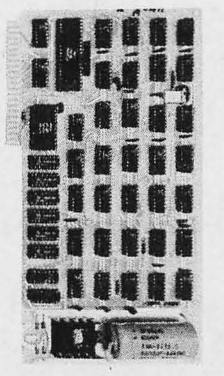
COMMENTARY: At this writing (10-1-77), the ESAT-100 is the only Stand Alone Terminal board requiring only black and white TV set and ASCII Keyboard. You do not have to have a S-100 Bus Machine, or even a computer. May be used in conjunction with a Modem and your home TV set to provide a timeshare type terminal at any Baud rate you desire.

Note, commercial terminals use an 80 character x 24 line format. However, we have chosen 32 character x 16 lines for television set applications because of the limited resolution available on most TV sets.

Nonetheless, for those of you who are the owners of either high quality video monitors, or the best of Japan's TV receivers, we offer the:

Scrollboard Adapter Kit (designed to fit on ESAT-100) with 64 characters x 16 lines and Automatic Scrolling for $29.95

M & R Supermod R F Modulator for antenna connection with your TV set. Runs off of ESAT 100 power supply color and black & white for $24.95



CAP. SPECIAL:
The Highest Quality,
By-pass anywhere:
VK06 0.1/100V for $0.29

Full Color TV Game Kit:
Includes MM57100 Game Chip, MM53104 Clock Gen., LM1889 N Color Modulator, 3.58 MHz color burst oscillator crystal, variable cap for crystal adj. and PC board. Very impressive on color sets. All data sheets and schematics included...$25.95

Direct Reading Capacitance Meter Kit:
Here is a handy shop item incorporating a reliable and accurate measurement method which works from 01 pico + d to 10 micro + d. Includes large 6" scale analog meter movement, all electronic parts and full documentation and theory of operation. Requires 5V and +-12V power. Makes accurate readings directly and instantly! No tuning, fiddling or interpolation required. Full documentation and theory...$39.95

5V Regulated Power Supply Kit:
Provides regulated 5V @ approx. 250 ma for benchwork, breadboards and small projects. Includes LM340T-5 regulator, rectifiers, filter caps, and wall-plug transformer. Full notes...$6.95

Nicad Batteries and Charger:
2 ea 2AH 'C' size Nicads (great for memory power-fail CKTS, cold weather flashlights, etc. and charger...$13.95

LINEAR

LM380N	1.39
LM340T-5,12,15	.99
NE560	2.95
NE561	2.95
NE565	1.95
NE566	1.49
NE567	1.49
LM1812	4.95
LM1889	4.95

CPU SPECIALS

8080A	10.95
Z80 (2mHz) incl.	
18mHz Xtal	33.95
Z80A (4mHz) incl.	
36mHz Xtal	39.95

OPTOCOUPLERS

4N26	1.00

PROMS

8223 Special	10/9.95
82S115 8 x 512	15.95

EPROMS

2708	650nS	12.95
2708	450nS	16.95
C1702A	1000nS	4.95
MM5203Q		4.95
MM5204Q		9.95

8K Static Memory Boards
Tested, Burned-In, Complete
$179.00 for 450nS
$239.00 for 250nS
$ 19.00 for Bare Board

MERRY CHRISTMAS!

VERBATIM Removable Magnetic Storage Media
PRICE REDUCTION!

	1-9	10-25	26-100
Minidiskettes	4.79	4.65	4.45

MD525-01 (Soft Sector) for: Intelligent Systems, Magnovox, Microkit, and Vector Graphics
MD525-10 (10 Sector, Hard) for: Digi-log, North Star, Polymorphic, Tei Inc., and Wang
MD525-16 (16 Sector, Hard) for: Altair, Comtek, Micropolis, R2E, Realistic Controls, and Teleray- Research Inc.

	1-9	10-25	26-100
Standard Size Diskettes	5.99	5.33	4.79

FD34-1000 (Soft Sector, IBM Std.)
FD32-1000 (Hard Sector, Inner dia.)
FD65-1000 (Hard Sector, Outer dia.)

	1-9	10-25	26-100
Cassettes R-300 Digital Direct	5.25	4.99	4.35

RAMS

21L02-1	450nS	1.25
21L02	250nS	1.50
2111AL-4	450nS	2.30
C3107B	60nS	1.99
PD411D-4	135nS	3.99

and OTHERS

AY5-1013	4.95
MM5369	1.69
MM5320	6.00
2513 Up.	5.95
2513 Lo.	6.95
IN4005	11/1.00
IN4148	15/1.00

Electrolabs
POB 6721,
Stanford, CA 94305
415-321-5601

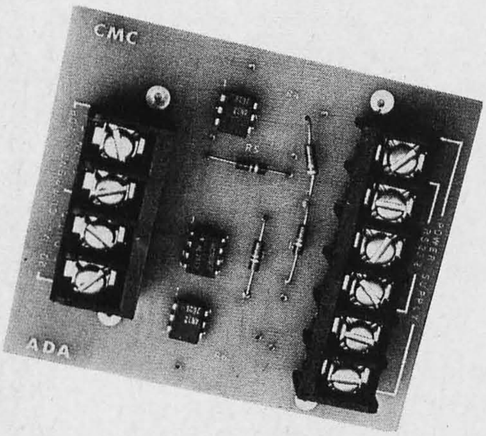
SATISFACTION
100% GUARANTEED!

TO ORDER:
Send check or money order and include $1.00 for shipping, 1.00 (optional) for insurance, and please include 6½% sales tax if you are a California state resident. COD orders add .85 Thank you.

Circle 48 on inquiry card.

What's New?

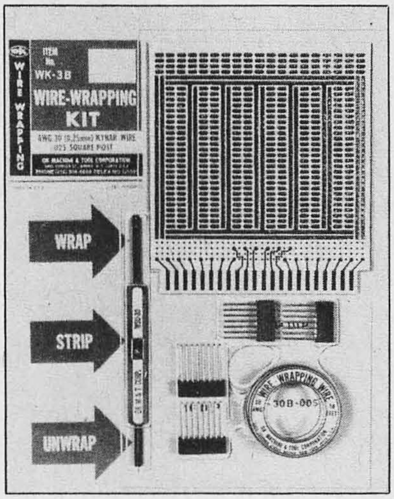
In Isolation, RS232 and 20 mA Get Along



This adapter provides both RS232 to 20 mA current loop and 20 mA to RS232 conversion, implemented with optoisolators. It can be used to interface a terminal of one type to a computer of the other type for both input and output, or it can be paralleled to provide secondary output on a Teletype or RS232 printer while still using the computer's primary terminal. The adapter comes assembled and tested with user instructions. It is built on a 3 by 3.5 by 1 inch (7.6 by 8.9 by 2.5 cm) printed circuit board, with drilled, plated-through solder pads for all connections priced at $24.50, or with barrier strips and screw terminals (pictured) for $29.50. Contact Connecticut Microcomputer, Pocono Rd, Brookfield CT 06804.■

Circle 530 on inquiry card.

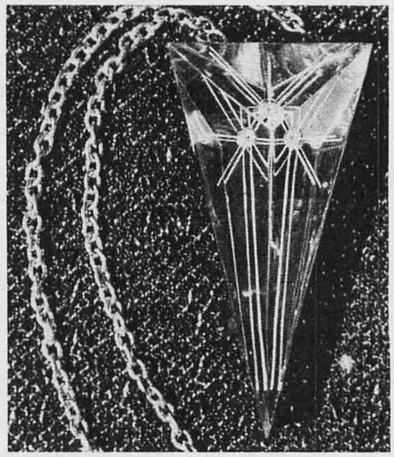
This Wire Wrapping Kit is OK



Hobbyists and prototype engineers can get started with wire wrapping using this new kit. The kit features a combination tool for wrapping, stripping and unwrapping, a 50 (15.2 meters) foot roll of wire, two 14 pin and two 16 pin DIP wire wrap sockets, and a 4 by 4.5 inch (10.2 by 11.4 cm) prototyping board with a 44 pin edge connector using standard .156 (0.4 cm) inch spacing. The Model WK-3B kit is available for $15.95 from your local electronics outlet or directly from O K Machine and Tool Corporation 3455 Conner St, Bronx NY 10475, (212) 994-6600.■

Circle 532 on inquiry card.

Give a Jewel

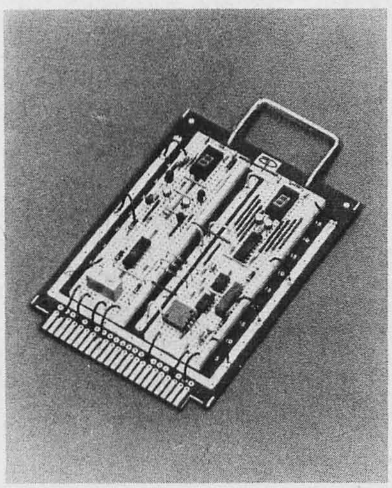


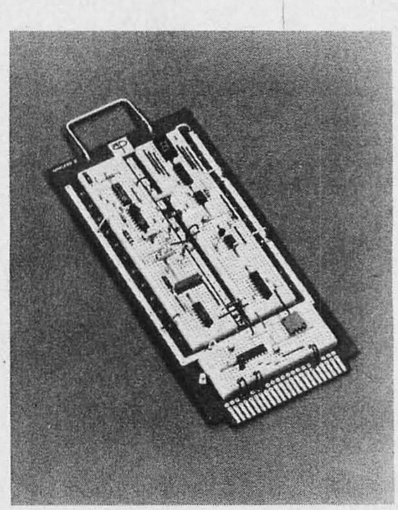
At last, integrated circuit technology has found an imaginative use in jewelry. The Star Jewel pendant has a red light emitting diode in a mirrored multi-faceted lucite setting. An integrated circuit, powered by two inexpensive hearing aid batteries, blinks the LED about three times a second. One set of batteries will run the circuit for about 300 hours, which should be enough for two to three months of use, depending on how vigorous your night life is. Star Jewels are available with red, green, blue, silver and smoke colored gems and come complete with pendant chain, batteries and a velveteen pouch. The price of $31.25 includes postage and insurance, and the electronics and wokmanship are guaranteed for one year. A catalog of hand crafted jewelry, ray guns, and science fiction and computer art is also available for $.25 (free with a Star Jewel order) from ATRA, POB 456, Minneapolis MN 55440.■

Circle 533 on inquiry card.

Improvements in Solderless Breadboarding

New versions of A P Products' Unicards provide five tie points rather than four in each row of terminals, making them ideal for breadboarding 24 and 40 pin LSI integrated circuits. The Unicards provide solderless, plug-in tie points in a .1 by .1 inch matrix. They have a standard .156 inch center spacing 22 pin double sided edge connector and plug into standard 5.25 inch card racks. Extender cards are also available. Additional features include rubber feet for bench work and extractor handles for easy withdrawal from card racks. Unicard I, priced at $31.50, has 960 tie points, while Unicard II has 1620 tie points. The cards are available from A P Products, 72 Corwin Dr, Box 110, Painesville OH 44077, (216) 354-2101.■





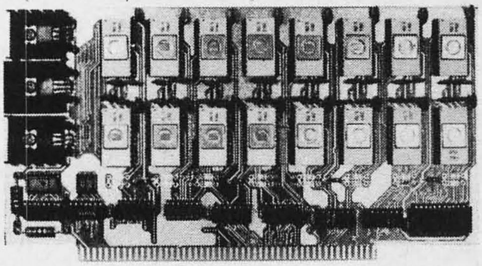
Circle 531 on inquiry card.

Digital Research Corporation

16K E-PROM CARD

$69.95 (KIT)

S-100 (IMSAI/ALTAIR) BUSS COMPATIBLE



DEALER INQUIRES INVITED

IMAGINE HAVING 16K OF SOFTWARE ON LINE AT ALL TIME!

KIT FEATURES:

1. Double sided PC Board with solder mask and silk screen and Gold plated contact fingers.
2. Selectable wait states.
3. **All** address lines and data lines buffered!
4. All sockets included.
5. On card regulators.

USES 2708's!

KIT INCLUDES ALL PARTS AND SOCKETS! (EXCEPT 2708's)

ADD $25 FOR ASSEMBLED AND TESTED

SPECIAL OFFER: Our 2708's (650 NS) are **$12.95** when purchased with above kit.

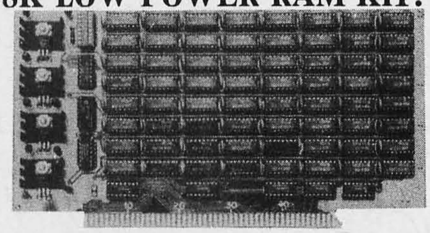
FULLY STATIC! $149.00 KIT

KIT FEATURES:

1. Double sided PC Board with solder mask and silk screen layout. Gold plated contact fingers.
2. All sockets included!
3. **Fully** buffered on all address and data lines.
4. **Phantom** is jumper selectable to pin 67.
5. **FOUR** 7805 regulators are provided on card.

S-100 (IMSAI/ALTAIR) BUSS COMPATIBLE

8K LOW POWER RAM KIT!



USES 21LO2-1 RAM'S.

PRICE WAR!
For a limited time only:
Buy two 8K Kits for **$129 ea.**

Fully Assembled and Burned In — $179.00

Blank PC Board With Documentation — 29.95

TAKE THAT BILL GODBOUT!

COMPUTER GRADE CAP.
48,000 MFD 25 WVDC Mallory
$3.95 **NEW!**

TR1602B UART
$4.50

8 POSITION DIP SWITCH
By Cts. Fits 16 Pin Socket.
$1.95

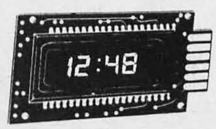
RCA HOUSE #2N3772
NPN Power Transistor. 30 AMP. 150 W. VCEO-60. TO-3. Vastly out performs 2N3055. Reg. List $3.04
2 FOR $1

T. I. ASCII CHARACTER GENERATOR
TMS 4103 JC. 28 PIN CER DIP. Has seven bit COLUMN Output for use with Matrix hard copy devices. With specs.
$3.50

NEW! **4K STATIC RAM'S** NEW!
2114. The industry standard. 18 PIN DIP. Arranged as 1K X 4. Equivalent to FOUR 21LO2's in ONE package! TWO chips give 1K X 8, with data.
2 FOR $24 **450 N.S.!**

MOTOROLA 7805R VOLTAGE REGULATOR
Same as standard 7805 except 750 MA OUTPUT. TO-220. 5VDC OUTPUT.
$ .44 each **10 FOR $3.95**

NATIONAL SEMI. MA1003 CAR CLOCK
Not a kit. Complete tested module. Works on 12 VDC, has on board time base. Sold by others at $24.95. Big .30" Bright Green Digits. Same as used by Detroit in new cards.



$19.95

EDGE CONNECTOR — $1.50

Z - 80 PROGRAMMING MANUAL
By MOSTEK, the major Z - 80 second source. The most detailed explanation ever on the workings of the Z - 80 CPU CHIPS. At least one full page on **each** of the 158 Z - 80 instructions. A **MUST** reference manual for any user of the Z - 80. 300 pages. Just off the press! A D.R.C. exclusive! **$12.95**

2708 1KX8 EPROMS 2708
Prime new units from a major U.S. mfg. 650 N.S. access time. Equivalent to four 1702A's in one package!
$15.75 each

TERMS: ORDERS UNDER $15 ADD $ .75. NO C.O.D. WE ACCEPT VISA, MASTER CHARGE AND AMERICAN EXPRESS CARDS. MONEY BACK GUARANTEE ON ALL ITEMS. TEXAS RESIDENTS ADD 5% SALES TAX.

Digital Research Corporation

(FORMERLY DRC) NOT AFFILIATED WITH ANY OTHER MAIL ORDER COMPANY.

P. O. BOX 401247 • GARLAND, TEXAS 75040 • (214) 271-2461

WE PAY POSTAGE!

Circle 42 on inquiry card.

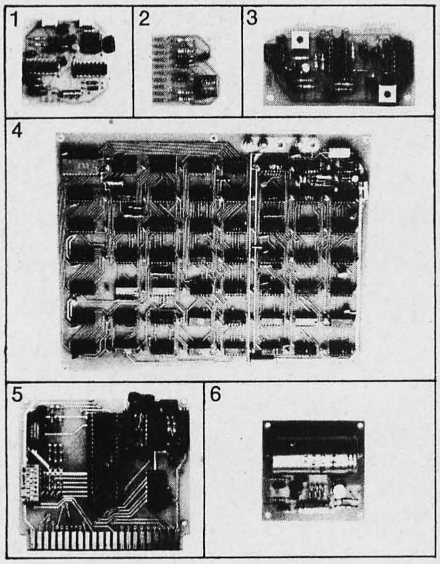
Hobby Computer Kits ●●●

1 MODEM Part no. 109

Type 103
Full of half duplex
Works up to 300 baud
Originate of Answer
No coils, only low cost components
TTL input and output
Connect 8 ohm speaker and crystal mic. directly to board
Uses XR FSK demodulator
Requires +5 volts
Board only $7.60, with parts $27.50

2 RS-232/TTL INTERFACE Part no. 232

Converts TTL to RS-232, and converts RS-232 to TTL
Two separate circuits
Requires +5 and −12 volts
All connections go to a 10 pin gold plated edge connector
Board only $4.50, with parts $6.00



3 TAPE INTERFACE Part no. 111

Play and record Kansas City Standard tapes
Converts a low cost tape recorder to a digital recorder
Works up to 1200 baud
Digital in and out are TTL
Output of board connects to mic. input of recorder
Earphone of recorder connects to input on board
Requires +5 volts, low power drain
No coils
Board only $7.60, with parts $27.50

4 TELEVISION TYPEWRITER Part no. 106

Stand alone TVT
32 char/line, 16 lines, modifications for 64 char/line included
Parallel ASCII (TTL) input
Video output
1K on board memory
Output for computer controlled curser
Auto scroll
Non destructive curser
Curser inputs: up, down, left, right, home, EOL, EOS
Scroll up, down
Requires +5 volts at 1.5 amps, and −12 volts at 30mA
Board only $39.00, with parts $145.00

5 UART and BAUD RATE GENERATOR Part no. 101

Converts serial to parallel and parallel to serial
Low cost on board baud rate generator
Baud rates: 110, 150, 300, 600, 1200, and 2400
Low power drain +5 volts and −12 volts required
TTL compatible
All characters contain a start bit, 5 to 8 data bits, 1 or 2 stop bits and either odd or even parity
All connections go to a 44 pin gold plated edge connector
Board only $12.00, with parts $35.00

6 RF MODULATOR Part no. 107

Converts video to AM modulated RF, Channels 2 or 3
Power required is 12 volts AC C.T., or +5 volts DC
Board only $4.50, with parts $13.50

4K/8K STATIC RAM Part no. 300

8K Altair bus memory
Uses 2102 Static memory chips
2-4K Blocks
Blocks can be addressed to any of 16 4K sections
Vector input option
TRI state buffered
Board only $22.50, with parts $160.00

TIDMA Part no. 112

Tape Interface Direct Memory Access
Record and play programs without bootstrap loader (no prom)
Has FSK encoder/decoder for direct connections to low cost recoder at 625 baud rate, and direct connections for inputs and outputs to a digital recorder at any baud rate
S-100 buss compatible
Comes assembled and tested for $160.00

APPLE 1 MOTHER BOARD Part no. 102

10 slots −44 pin (.156) connectors spaced ¾ inch apart
Connects to edge connector of computer
Pin 20 and 22 connects to X & Z for power and ground
Board has provisions for by-pass capacitors
Board costs $15.00

D. C. POWER SUPPLY Part no. 6085

Board supplies a regulated +5 volts at 3 amps., +12, −12, and −5 volts at 1 amp
Board has filters, rectifiers, and regulators
Power required is 8 volts AC at 3 amps., and 24 volts AC C.T. at 1.5 amps
Board only $12.50

TO ORDER

Mention part number and description. For parts kits add "A" to part number. Shipping paid for orders accompanied by check, money order, or Master Charge, BankAmericard, or VISA number and signature. Shipping charges added to C.O.D. orders, Calif. res. add 6.5% for tax. Parts kits include sockets for all ICs, components, and circuit board. Documentation is included with all products. Dealer inquiries invited.

ELECTRONIC SYSTEMS

P.O. Box 212, Burlingame, CA 94010
(408) 374-5984

APPLE II I/O BOARD KIT

Plugs Into Slot of Apple II Mother Board

FEATURES:
- 1 8 Bit Parallel Output Port (Expandable to 3 Ports)
- 1 Input Port
- 15mA Output Current Sink or Source
- TTL or CMOS Compatible
- Addressable anywhere in memory output area
- Can be used for peripheral equipment such as printers, floppy discs, cassettes, paper tapes, etc.

KIT INCLUDES:
P.C. Board, I.C.'s Sockets and Assembly Manual.

PRICE:
1 Input and 1 Output Port for $49.00

1 Input and 3 Output Ports for $64.00

DEALER INQUIRIES INVITED

UNGAR SOLDERING IRONS

27W SOLDERING IRON KIT	I.C. DESOLDERING KIT	10W ASSEMBLED SOLDERING IRON
Includes iron, 2 tips, roll of solder and iron stand	Includes iron and 3 desolder tips for dual in lines, cams, etc.	
$6.72	$24.70	$14.97



LD-130 3 DIGITS A/D CONVERTER

$11.95

PUSH BUTTON SWITCH Red or Green

3 for $1.00

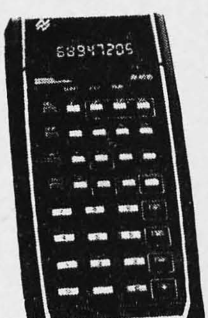


A FULLY PROGRAMMABLE SLIDE RULE

the MATHEMATICIAN with 100 STEPS

$29.95

• RPN logic with "built-in" hierarchy for increased speed and accuracy in calculating sequences involving arithmetic, trigonometric, logarithmic, power or exponential functions. • A three-level stack plus seperate accumulating memory for quick, accurate solutions to complex calculations. • Eight-digit LED display with full-floating decimal system. • Common and natural logarithms and antilogarithms. • Sine, cosine, tangent and inverse trigonometric functions. • Instant automatic calculation of powers and roots. • Instant conversions of radians to degrees or vice versa. • Square, square root, and reciprocal calculations. • Pi, change sign, and register exchange keys. • Automatic reciprocals. • Ability to automatically sum squares. • Storage memory. • "Roll-down" clear. • MOS/LSI solid-state circuitry. • Engineered and manufactured by National Semiconductor Corp., a world leader in solid-state technology.



• Simplified programming. You simply engage a learn switch and perform a problem in normal manner. The 4615 records the formula and lets you debug the program as it's written. • The learn-mode capacity total 100 separate steps. • Several different programs can be contained at the same time. • Constant factors can be entered as program steps. • Delete feature lets you correct programs while you are writing them. • Skip key permits skipping over entire programs to access additional programs within 100-step capacity. • Programs remain intact until new programs are written over or until your 4615 is turned off. • You have total freedom to select keyboard entries as variables or constants. • Automatic warning signal in display lets you know when you exceed programming capacity. • The 4615 is rechargeable and comes complete with nickel cadmium batteries.

10 DAY MONEY BACK GUARANTEE

___ 4615 @ 29.95 ________
___ AC charger @ 4.95 ________
___ Carrying case @ 2.95 ________
___ Calc. stand @ 1.95 ________

3RD GENERATION **ONLY $63.00**

ASCII KEYBOARD KIT



FURTHER IMPROVEMENTS, MORE FEATURES

- TTL Logic Circuits
- Power: +5V, 275mA
- Upper and Lower Case
- Full ASCII Set (Alpha Numeric, Symbols, Control)
- 7 ot 8 Bits Parallel Data
- Optional Serial Output
- Selectable Positive or Negative Strobe, and Strobe Pulse Width
- 'N' Key Roll-Over
- Full Debounced
- Carriage Return Key
- Repeat Function Key
- Shift Lock, 2 Shift Keys
- 4 User Defineable Keys
- P.C. Board Size: 17-3/16" x 5"

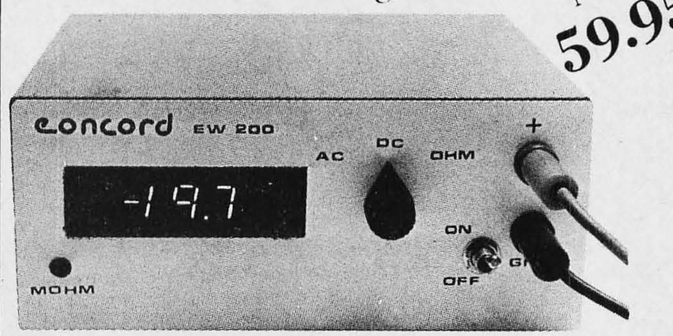
OPTIONS:
- Metal Enclosure (Painted IBM Blue and White $25.00
- 18 Pin Edge Con. $2.00
- I.C. Sockets ... $4.00
- Serial Output (Shift Register) $2.00
- Upper Case Lock Switch (for Capital Letters and Numbers) $2.00

KIT NUMBERS: Keyboard, P.C. Board, all required components and assembly manual.

NOTE: If you have this 63 Key Teletype Keyboard you can buy the Kit without it for $44.95.

CHRISTMAS SPECIAL

LSI DMM KIT Reg. $77.77 NOW 59.95



- AUTO RANGING • AUTO POLARITY
- AUTO ZERO • 3 LARGE DIGITS (½")
- RECHARGABLE

MEASUREMENT RANGES: Voltage (AC-DC) 1MV-100V; Current (AC-DC) 10μA-1A; **RESISTANCE:** 1Ω-10MΩ. Basic DC accuracy, better than 0.1% ± 1 digit. **POWER:** 4 AA batteries (rechargable batteries optional).

• NI-CAD BATTERIES: $6 • AC CHARGER: $4.95
• ENCLOSURE: $12.95 • TEST LEADS: $2.95
• SHUNT KIT FOR 3 CURRENT RANGES: $4.75

KEYBOARD, DMM and CALCULATOR: $3.50.
ALL OTHERS – $1.00.
California residents add 6% sales tax

ELECTRONICS WAREHOUSE Inc.

1603 AVIATION BLVD.
REDONDO BEACH, CA. 90278
TEL. (213) 376-8005
WRITE FOR FREE CATALOG
You are invited to visit our store at the above address

Circle 51 on inquiry card.

Everyone Should Learn Programming, TI Calculates

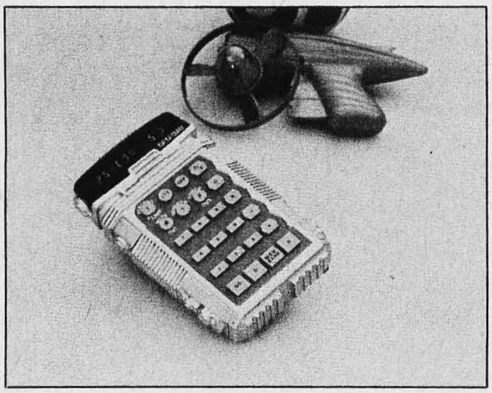


Photo 1: TI's DataMan offers arithmetic practice problems and math strategy games for the youngster. It includes a "beat the clock" timer feature and a "whiz-bang" display for correct answers.

Circle 555 on inquiry card.

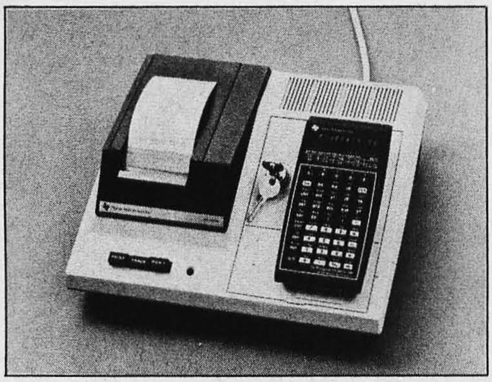


Photo 2: The TI 58, shown mounted on the PC-100A printing cradle with alphabetic printing capability.

Circle 556 on inquiry card.



Photo 3: The ultra compact DataClip calculator will run for 1000 hours on one set of batteries.

Circle 557 on inquiry card.

In a move calculated to encourage interest in programming and computational problem solving, Texas Instruments introduced a number of new general purpose and specialty calculators recently. In a press announcement, TI noted that nearly 400,000 people purchased programmable calculators in 1976, and estimated that over 3 million people a year would be buying programmables by 1979. Included among TI's new products are calculators with "Solid State Software" program libraries, new tutorial books on programming, and a new teaching device for the kiddies.

At the top of the new line are the TI Programmables 57, 58 and 59. The Programmable 57 ($79.95), which evidently supersedes the SR-56, includes a 150 keystroke program memory, insert and delete keys for editing programs, and ten labels for "relocatable" program branching. The 57 comes with a new learning guide, *Making Tracks Into Programming,* which replaces the usual owner's manual.

The Programmable 58 ($124.95) and 59 ($299.95) feature larger program and data memories, and plug-in Solid State Software read only memory modules containing up to 5000 program steps each. The ROM libraries range from applied statistics and surveying to real estate, investment, aviation and marine navigation programs.

Internal memory on the 58 and 59 can be allocated either to program steps or to data registers. The unit of allocation is ten data registers or 80 program steps. The TI 58 has up to 480 program

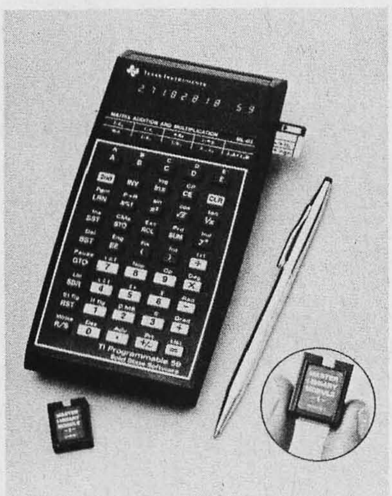


Photo 4: The TI 59 features Solid State Software plug-in program library modules of up to 5000 program steps, as well as a magnetic card reader and internal memory for 960 program steps.

Circle 558 on inquiry card.

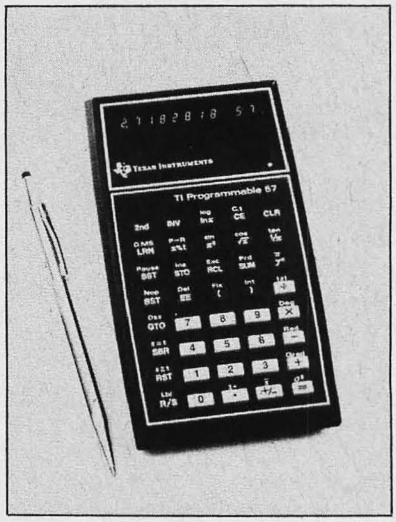


Photo 5: The new TI 57 programmable calculator can store up to 150 keystrokes.

Circle 559 on inquiry card.

steps or up to 60 data registers, while the TI 59 has up to 960 program steps or 100 data registers. (When all 100 memory registers are in use on the TI 59, 160 program steps are still available.) The TI 59 also has a magnetic card facility. Using two cards, up to 960 program steps can be recorded and reloaded in this way.

Additional features of the TI 58 and 59 are up to ten registers for looping, incrementing and decrementing, up to ten flags which can be set, reset or tested, and up to six levels of subroutine calls. Program steps may be addressed in absolute, indirect and label modes, while data registers may be addressed directly or indirectly.

A related new product is the PC-100A printing cradle, which can be used with any TI programmable calculator except the Programmable 57. This printer has 64 alphabetic, numeric and special characters which can be printed at the rate of 60 characters per second. Up to 20 characters can be printed per line on 2.5 inch (6.4 cm) wide thermal paper. The PC-100A can also be used to print, list or trace program steps as an aid to debugging. It is priced at $199.95.

Another element in TI's consumer education program is a new book, *Calculating Better Decisions.* Priced at $4.95, the book is offered in a package with the SR-51-11 calculator for $69.95. The book concentrates on use of the SR-51-11's advanced statistical functions in psychology and the social sciences and in business finance.

One strategy in TI's plan to place several calculators in every home is to design specialty calculators for specific types of consumers or consumer uses. A good example of this strategy is the new MBA calculator, aimed at the

Continued on page 206

Circle 53 on inquiry card.

An Exotic Way to Tell Time

For the man or woman who has everything, Hewlett-Packard now offers the most exotic wristwatch ever designed, the HP-01. Priced at a cool $650 in stainless steel or $750 with a gold filled case, the HP-01 is an integrated timepiece and calculator. It has 28 keys (six finger operated, 22 stylus operated) and a light emitting diode (LED) display with 12 different display modes or indicators.

The HP-01 displays the time of day in either 12 or 24 hour format, with an indicator to show PM time. The user can add or subtract time, with automatic date adjustment if the time change is across midnight. Time can also be converted to and from decimal hours. The device has two alarms, one which can be set for a specific time and one based on elapsed time as tracked by an internal

Continued on page 208



Continued from page 204

hundreds of thousands of masters in business administration students and degree holders. The MBA ($79.95), like the TI Business Analyst, has special keys to accomplish common business functions such as net present value, internal rate of return, trend line analysis, mean, variance and standard deviation, accumulated interest and remaining loan balance, and annuity calculations. But it also has a simplified programming feature, allowing storage of up to 32 program steps but no tests and branches, as well as 12 data memories.

Another new product, the TI 1680 "Replay Calculator," is designed specifically for applications such as checkbook reconciliation or "balancing the books" in an accounting system. The TI 1680 ($29.95) allows the user to recall up to 20 previously entered numbers and arithmetic operations, and to change the previous entries to see what effect the change has on the calculated result.

If the typical TI calculator is too heavy or bulky for your pocket, TI still has something for you. The new Data-Clip ($34.95) is about the size of a 6 inch (15.2 cm) ruler, and no thicker than a pencil. It has five functions and an 8 digit liquid crystal (LCD) display, and will operate up to 1000 hours on a set of batteries.

And finally, for the kiddies, there's DataMan! TI evidently believes that youngsters should get started with calculators at an early age (so that they'll be ready for programming by age 12?). DataMan ($24.95), patterned after the highly successful "Little Professor" introduced a year ago, offers children practice with arithmetic problems and fun with math strategy games. Learning activities possible with the unit include the Answer Checker, Problem Storage, Math Tables and Missing Numbers, and games include Wipe Out and Force Out. A special "beat the clock" timing feature adds to the fun and challenge of the exercises and games. Correct answers are rewarded with "whiz-bang," a highly

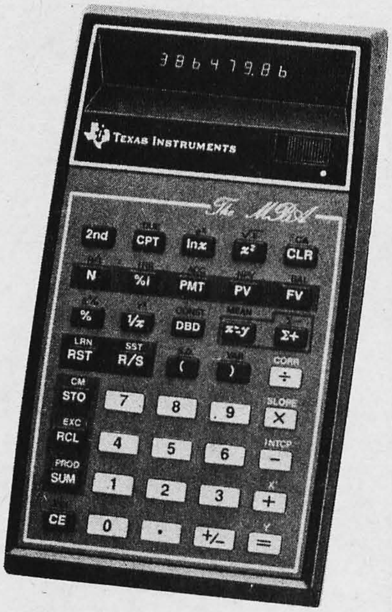


Photo 6: The new MBA calculator is aimed at business administration students.

Circle 560 on inquiry card.



Photo 7: This new book is available alone or packaged with the SR-51-11 calculator.

Circle 562 on inquiry card.



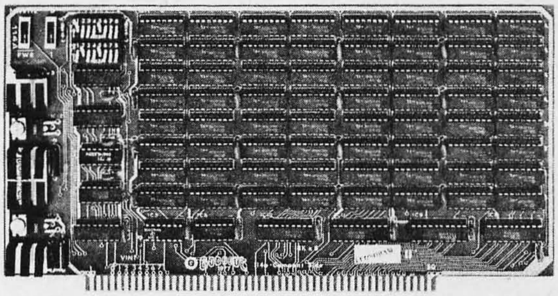
Photo 8: The TI 1680 replay calculator, which remembers the last 20 numbers and operations entered, is useful for checkbook balancing.

Circle 561 on inquiry card.

visual action packed display "on the order of modern stadium scoreboards." DataMan can be adjusted to present problems appropriate to the youngster's achievement level. It comes with a math activity book and helpful hints for parents.

Clearly, TI's calculators are moving in the direction of increasing diversity and sophistication. In fact, the TI 59 with the PC-100A printing cradle probably qualifies as a personal computer, with limited alphabetic data handling, hard copy, and mass storage. But when will TI introduce a product similar to the Commodore PET or the Radio Shack TRS-80, and what impact will it have on the marketplace? This is anyone's guess. In the meantime, you can find out about today's calculators by contacting Texas Instruments Inc, Inquiry Answering Service, POB 53 (attn: the product of your choice), Lubbock TX 79408.■

The 4th Annual
HOLIDAY SPECIAL
8K Econoram II T.M. -- 4 for $475.00 !

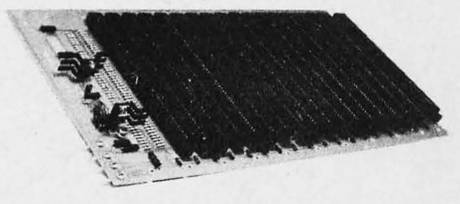


Under $0.0018 per bit! Not only are we celebrating the holiday season...but we're answering a challenge from Billy Gage and Jim Tanner at DRC. There's been a bit of friendly rivalry lately over who can deliver the most for the least; we have always prided ourselves on being able to do just that, and with 32K of quality memory for $475.00, we think even DRC is going to have to agree with us.

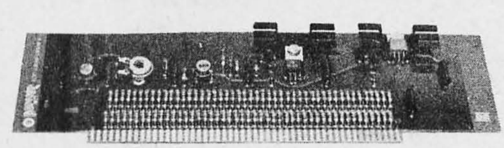
But price is by no means the only reason to buy a Godbout memory board: those who know memory appreciate the many options packed into a basic memory board. Extras like a vector interrupt provision if you try to write into protected memory. Configuration as two independent 4K blocks (both individually protectable). A selectable write strobe for either PWR or MWRITE. An all static design. The ability to handle DMA devices. Guaranteed speed under 450 ns (with on board wait state logic for use with 4 MHz Z-80) and guaranteed current under 1.5A (1250 mA typ). All inputs and outputs fully buffered; outputs are tri-state for use with bi-directional busses. And of course, sockets for all ICs, legended board with solder mask, low power Schottky support ICs, assembly instructions, a 1 year warranty on all parts...**this isn't just another board, this is a board you can depend on.**

Prices apply to boards in kit form only.

MOTHERBOARD
10 Slots $90 18 Slots $124

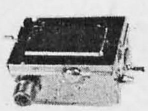


Includes **all** edge connectors, plus active terminations to minimize crosstalk, noise, overshoot, and ringing that may be present with unterminated busses. Excellent for stand alone system, or add to existing systems. Kit form only.



Active Terminator $29.50

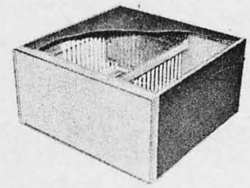
Plugs into any S-100 Motherboard whose buss lacks active terminations. Cleans up noise, crosstalk, overshoot, and other buss problems that can scramble data unpredictably. Kit form only.



video modulator

Video and 5 V DC in, channel 3 and 4 out.

$7.50

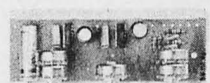


COMING SOON

CAVE grafix

2 WATT STEREO AMP

Assembled stereo 2W amp uses LM380s. With volume, tone, balance controls. Use with 12V for portable amp or auto use. **Only $10.00**



Vector VP2 Assembled Microcomputer Case

This adjustable packaging system for S-100 buss microcomputers is compatible with Altair 8800 and IMSAI 8080 size cards. Outside, it is beautiful...with a dark blue textured vinyl finish and lines unmarred by external screws or fasteners. Inside, there is space for 21 cards total (on 0.75" centers) with a fully adjustable interior card mounting system (card guides and hardware provided for 12 cards). The interior is instantly accessible, the rear and front panels are removable and recessed.

And the clincher: our 10/11 slot motherboard fits perfectly inside this case. So if you want a classy home for your micro, look this over...it is the best we've seen.

VP2 ASSEMBLED MICROCOMPUTER CASE....$134.30

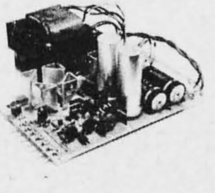
ECONORAM VI T.M.
12K for $235

The Heath H8 computer looks like a winner...and we'll be happy to supply you with **12K of memory for $235**, which happens to be what Heath charges for their 8K memory. All the same features as our popular 8K ECONORAM II tm kit (see above), with all static design, buffering, switched protect and phantom, and our 2 block configuration gives the ability to place 4K on 4K boundaries.

As usual, you get a legended, solder masked board, sockets for all ICs, top performance, and a warranty on parts.

Check out our memory if you want your H8 to get the most for the least.

1.8432 MHz xtal HC6 size with wire leads $5.95



Small System Power Supply

Finally, a quality, cost-effective supply for small systems. Gives you a full 4 Amps at 5 Volts, with crowbar overvoltage protection, along with half an Amp of +12 and half an Amp of -12...and an additional 10 mA supply, adjustable over 5 to 10V for biasing required by some CPUs. Although intended to be used with computer systems, it's also a dandy little bench supply for digital experiments.

$50

Personal to Frank Tinius: watch this space, it's happening!

TERMS: Please allow up to 5% for shipping, more for supply & VP2; excess refunded. Prices good through end of magazine cover month. Californians add tax. CODs accepted with street address. For BankAmericard®/VISA®/Mastercharge® orders ($15 min) call **415-562-0636**, 24 hours.

GODBOUT

BILL GODBOUT ELECTRONICS
BOX 2355, OAKLAND AIRPORT, CA 94614

SEVERAL GOOD REASONS WHY YOU SHOULD HAVE OUR FLYER: 1) CMOS 2) LINEARS 3) MICROPROCESSORS 4) POWER SUPPLIES 5) RESISTORS 6) CAPACITORS 7) DISPLAYS 8) SOCKETS 9) VECTOR PRODUCTS 10) ENCLOSURES 11) ALL THE OTHER THINGS WE CAN'T FIT INTO THIS SPACE.

Circle 56 on inquiry card.

Continued from page 206

timer. The alarms generate a 2.5 second beeping tone.

A timer and stopwatch feature displays elapsed time in hours, minutes and seconds or in minutes, seconds and hundredths of a second. Elapsed time readings can be taken and one reading can be stored in memory. In combination with the calculator functions, this feature allows the user to perform dynamic calculations. For example, while the user is on the telephone, the HP-01 can be set to calculate how much the call is costing, updating the display every second.

A 200 year calendar feature can display the date in US or European format, and can be used to find past or future dates given the current date and the number of days. The number of days between two dates, the day of the week and the day of the year for any date can also be determined.

The calculator functions include add, subtract, multiply, divide, percentage, net amount, chain and repeat calculations. Four digits of precision are provided, with automatic conversion to and from scientific notation as needed. A memory key provides access to a memory register, and two more memory registers are used for intermediate results of calculations.

In classic fashion, Hewlett-Packard engineers found innovative solutions to a number of difficult design problems in producing this device. The HP-01 employs a hybrid assembly of six CMOS chips, drawing a total of 15 μW, one ten thousandth of the power required by the HP-35. The watch case had to be water resistant, yet still have a keyboard that perforated the front plate and an alarm signal that could penetrate the back. This was done by placing a thin rubber diaphragm backed by a conductive layer under the keys. When a key is pressed, the rubber diaphragm provides the springback, and the conductive layer makes a direct connection with a gold contact on the surface of the CMOS hybrid. This method has worked without failure in tests where the keys have been pressed over one million times. The alarm works by exciting the upper piezoelectric ceramic layer of a 2 layer plate. The vibrations in the ceramic layer resonate through the stainless steel second layer, which acts as a diaphragm. With the watch itself as a sounding board, the results are audible beeps.

How long before we scarcely notice when our watch beeps and says, "It's four thirty. Don't forget to make that phone call before five."? In the meantime, the HP-01 is available from Hewlett-Packard Corporation, 1501 Page Mill Rd, Palo Alto CA 94304, (415) 493-1501.■

Circle 563 on inquiry card.

RCA's New COSMAC Manual

A 28 page *Instruction Manual for the RCA COSMAC Microterminal CDP 18S021*, MPM-212, describing the installation and application of a hand held data terminal for microcomputer systems using the CDP1802 microprocessor, is now available from RCA Solid State.

The Microterminal, which is the size and shape of a pocket calculator, provides a means of controlling a COSMAC-based system and supplies hexadecimal IO capability. It is designed to interface directly with the COSMAC Evaluation Kit CDP18S020 support hardware, but it can be designed into user built systems to provide the same control, communications and debugging functions.

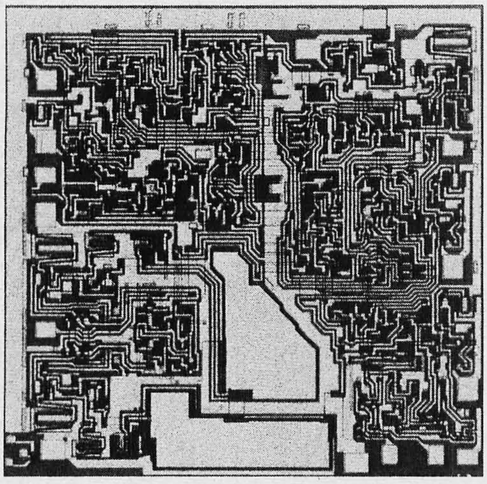
The manual includes a description of the hardware and the software programs available in the read only memory supplied with the Microterminal, and also includes installation instructions and utility program listings. The operating modes of the Microterminal are described along with several examples of operating sequences.

Copies of this new 8½ by 11 inch *Instruction Manual for the RCA COSMAC Microterminal CDP18S021*, MPM-212 may be obtained for $2 each (US price) from RCA Solid State Division, POB 3200, Somerville NJ 08876.■

Circle 565 on inquiry card.

Circle 76 on inquiry card.

New Clock Chip for 6800 Systems



Microcomputer designs based on the Motorola 6800 microprocessor and others such as the MOS Technology MCS6512 can be simplified with this monolithic clock generator chip. Requiring only a 5 V supply and a quartz crystal or RC (resistor and capacitor) network, the MC6875 provides buffered 2 phase clock outputs, automatic clock stretching for slow memories, direct memory access or dynamic memory refreshing. It also has request and grant logic for IO peripherals, and a Schmitt trigger power on and reset function. Available in a 16 pin dual in line package, the MC6875 costs $3.45 in 1000 piece quantities from Motorola Linear Products, 2200 W Broadway, Mesa AZ 85202, (602) 962-2294.■

Circle 534 on inquiry card.

Improved Version of 2708 from Intel

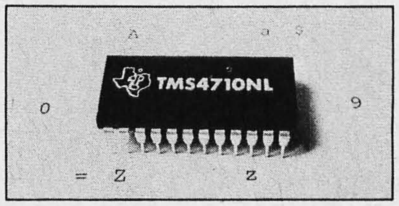
While the popular 2708 1 K byte EROM chip continues to drop in price, Intel has introduced an improved version of the chip, the 2758, which makes read only memory designs still simpler. The new chip is completely TTL compatible and requires only a 5 V power supply. It is directly upgradable to a 2 K byte 2716 EROM or a 2316E mask programmed ROM, both of which require only a 5 V supply. Power dissipation is less than 50% of the 2708, and a power switching input can be used to reduce dissipation by more than 80% when the chip is not selected.

Even more interesting are the device's programming features. The entire EROM can be programmed in less than a minute, twice as fast as the 2708, but in addition any number of bytes can be programmed at one time. And a byte is programmed simply by applying a TTL level signal to a program control input for 50 ms. The programming signal need not be pulsed but is simply direct current. These features make in-system programming much easier to design. (Who knows? Perhaps we'll see EROM boards with ultraviolet lights permanently mounted for periodic erasure and reprogramming.)

The 2758's introductory price of $26.60 is less than half the 2708's initial price of $65.50, and the 2758 and 2716 are expected to drop in price per bit much more rapidly than the 2708. The new devices have a head start on the production "learning curve" first traveled by the 2708. The 2758 is available from Intel Corporation, 3065 Bowers Av, Santa Clara CA 95051, (408) 246-7501.■

Circle 536 on inquiry card.

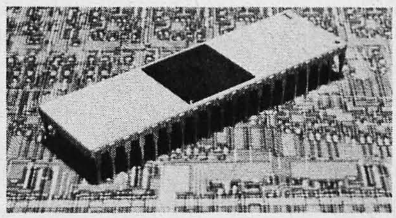
Full ASCII Character Set in a Byte Sized Dot Matrix



A new 1024 by 8 bit character generator ROM can be used to simplify video display and dot matrix printer controllers. The TMS4710 outputs dots for upper and lower case letters, numbers and special characters with spacing due to use of 5 by 7 characters in an 8 by 8 matrix. Three state outputs and dual output enables are provided in the plastic or ceramic 24 pin DIP package, which is priced at $10.66 or $12.66 in 100 piece quantities. Contact Texas Instruments' Inquiry Answering Service, POB 1443, M/S 669 (attn: TMS4710), Houston TX 77001, (713) 494-5115 ext 2781.■

Circle 537 on inquiry card.

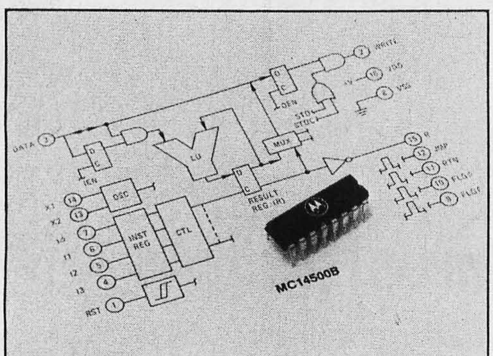
An 8 by 8 Multiply in 100 ns



Microcomputer applications such as real time music synthesis and speech processing, fast Fourier transform computations and fast floating point processors call for enormous numbers of multiplications each second. In these applications, a new chip from Monolithic Memories can help. It performs an 8 by 8 bit signed or unsigned 2's complement multiplication in 100 ns, yielding a 16 bit signed or unsigned result. The 40 pin bipolar LSI device uses a single 5 V power supply, consumes only 1 W and features three state outputs for pipelined operations. The MMI67558 is second sourced by ITT Semiconductors and sells for $64 in 100 up quantities from Monolithic Memories, 1165 E Arques Av, Sunnyvale CA 94086, (408) 739-3535.■

Circle 538 on inquiry card.

How About a 1 Bit Microprocessor?



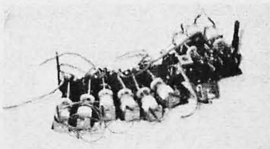
If eight bits just isn't right for your application, perhaps a 1 bit microprocessor will do the job. The Motorola MC14500B is a 1 bit static CMOS processor which performs logical operations on data occurring on a 1 bit bidirectional bus and data in a 1 bit accumulator. Five of the processor's 16 instructions provide Boolean functions of the data bus and accumulator bits, while the remaining instructions control data transfers and generate control signals.

Also called the Industrial Control Unit, the MC14500B is the monolithic embodiment of a programmable logic controller. With a looping program flow, the processor offers a simple way to sequentially control electronic and electromechanical devices in applications such as traffic controllers, office copiers, telephone dialing systems and microprogram control sequencers. The processor will operate at up to 1 MHz and includes an oscillator on the chip. The circuit, in a 16 pin ceramic or plastic DIP, is available for $7.58 in 100 up quantities from Motorola's Integrated Circuit Division, 3501 Ed Bluestein Blvd, Austin TX 78721, (512) 928-2600.■

Circle 535 on inquiry card.

Circle 135 on inquiry card.

CHRISTMAS SPECIAL





Selectric Terminal — USED, AS IS, but removed from working installation. Nationwide service available. $795.00

SPECIAL ITEMS FOR ABOVE TERMINAL
ASCII controller — 200 character buffer, all 128 characters, set at 110 to 1200 Baud (average throughput is 13 cps). ASR33 compatible RS232C interface. $225.00
BOTH UNITS ORDERED TOGETHER $890.00

SURPLUS BARGAIN:
New, in original containers. Double density floppy drives. Model GSI110. $549.00
2 for $1059.00

IBM format disk kit to use GSI110 drive with above terminal controller. $249.00
Board and EPROM only $99.00

Mastercharge, Visa or COD accepted.

$40.00 deposit on COD

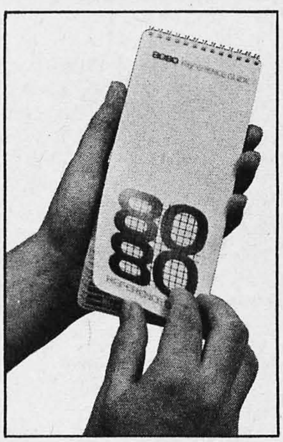
* LIMITED QUANTITIES Call or write by Dec. 5, 1977 to reserve for shipment on or before Dec. 12, 1977.

IMMEDIATE DELIVERY

Sharp & Associates Inc.
Box 10666, Edgemont Branch Golden, Colorado 80401

NEW 8080 and 8085 REFERENCE GUIDE

A TOTALLY NEW CONCEPT! SAVES TIME AND MONEY! MAKES YOUR JOB EASIER!



A **powerful** new tool for every serious 8080 user — professional and novice alike. Priceless **timesaver** for engineers, technicians, and programmers. Saves time and money in the lab, on the production line, or in the field.

Convenient pocket size — 3⅜ by 7¾ inches — gives quick and easy access to all vital reference data. No more searching here and there for codes, instructions, or definitions. It's all there — **at your finger tips** — everything you need to successfully use the 8080A and — Intel's new 8085 microprocessor.

Features cross listing, for rapid assembly and disassembly, of MACHINE CODES and MNEMONICS • Concise description of 8080 and 8085 OPERATIONS, SIGNALS, PINOUTS, and INSTRUCTIONS • Convenient cross conversion of OCTAL, HEXIDECIMAL, DECIMAL, ASCII, and EBCDIC codes • Easy-to-read tables of powers of two, eight, and sixteen . . . and much more . . .

Sturdy • Handsome • Easy-to-use • Data Packed

Your **timesaver** will give many years of professional service.

$12.95 each (plus postage & California sales tax) — 25% discount for 4 or more.

MoneyBack Guarantee: You must be fully satisfied or simply return the guide within 15 days for full and prompt refund.

URBAN INSTRUMENTS • 4014 CODY ROAD • DEPARTMENT C1 • SHERMAN OAKS • CALIFORNIA 91403

master charge THE INTERBANK CARD — BANKAMERICARD

PLEASE SEND ________ **8080 timesavers** to: CHECK/M.O. ENCLOSED ☐ BANKAMERICARD ☐ MASTER CHARGE ☐

NAME . . . . CARD NUMBER . . . .

STREET . . . .

CITY . . . . GOOD THRU . . . . 4 DIGITS ABOVE NAME . . . . (MASTER CHARGE)

STATE . . . . ZIP . . . . SIGNATURE . . . .

Circle 124 on inquiry card.

COMPU/TIME CT 100

COMPU/TIME offers A Real Darn Clever Enhancement to users of IMSAI/ALTAIR Microprocessors

S100 BUS COMPATIBLE

TIME & CALENDAR

COMPU/TIME	CT100	$199 Kit	$245 Assembled
COMPU only	C101	$149 Kit	$189 Assembled
TIME only	T102	$165 Kit	$205 Assembled
COMPU/TIME PC Board only			$ 80

TV-1

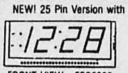
You will want to know about the TV-1 Video to Televisior Interface Kit.

No need to buy a separate Video Monitor if you already own a TV set. Just connect the TV-1 between your system video output and the TV set antenna terminals—that's all there is to it—to convert your TV set to a Video Monitor, and at a much lower cost!

PRICE $8.95

FCS 8000A — 3½ Digit — .8" Display

NEW! 25 Pin Version with colon & am/pm indicator



FRONT VIEW - FSC8000

SPECIAL

$4.95 EA.

- MAXIMUM FORWARD CURRENT — 25 mA.
- Connects almost one for one with 3817, 3817A or D. (3817 available at $5.00 each).
- Typical segment current 8mA except colon, 10 hrs. b & c and 10 min, a & d which are 16 mA.
- Forward voltage drop 1 5 volts

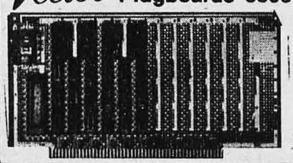
Vector

SLIT-N-WRAP WIRE WRAP TOOL

- Slits and opens insulation exposing bare wire
- No pre-cutting or pre-stripping.
- Comes complete with two - 100 ft spools #28 AWG wire

Model P180 $24.50

***Vector* Plugboards 8800V**



Universal Microcomputer/Processor plugboard, use with S-100 bus complete with heat sink & hardware 5.3 x 10 x 1/16"

$19.95

CHANNEL F™

FAIRCHILD VIDEO ENTERTAINMENT SYSTEM



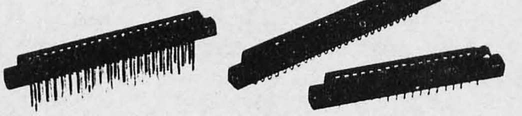
$159.95

- Freeze Action • Speed Option
- Automatic time and scorekeeping
- Battery-free AC operation
- Dual controls with 8-way action
- Built-in Pro Hockey and Tennis games
- Easy hook-up on any B/W or Color TV
- Factory warranty

Channel F — additional cartridges — $17.95 ea.

0811 — Tic-Tac-Toe/Shooting Gallery Quadra-doodle/Doodle/
0812 — Desert Fox/Shooting Gallery
0813 — Blackjack (1 or 2 players)
0814 — Spitfire (1 or 2 players)
0815 — Spacewar (2 players)
0818 — Magic Numbers (computer logic)
0819 — Drag strip (1 or 2 players)
0820 — Maze (2 players)
0822 — Baseball (2 players)

EDGE CONNECTORS



NO. PINS		TYPE		
20	DUAL	10 PIN	GOLD	$ .50
30	DUAL	15 PIN	GOLD	.75
44	DUAL	22 PIN	GOLD	1.95
44	DUAL	22 PIN	GOLD	2.50
80	DUAL	40 PIN	GOLD	4.95
86	DUAL	43 PIN	GOLD(6800)	5.00
100	DUAL	50 PIN	GOLD(IMSAI/ALTAIR)	4.25
100	DUAL	50 PIN	GOLD(IMSAI/ALTAIR)	4.95
100	DUAL	50 PIN	GOLD(NO EARS-IMSAI	3.50
100	DUAL	50 PIN	TIN (.1" SPACING)	3.25

ALTAIR/IMSAI CARD GUIDES .25¢ EACH

7400N TTL

SN7400N	.16	SN7459A	.25		
SN7401N	.16	SN7460N	.22	SN74154N	1.00
SN7402N	.21	SN7470N	.45	SN74155N	.99
SN7403N	.16	SN7472N	.39	SN74156N	.99
SN7404N	.18	SN7473N	.37	SN74157N	.99
SN7405N	.24	SN7474N	.32	SN74160N	1.25
SN7406N	.20	SN7475N	.50	SN74161N	.99
SN7407N	.29	SN7476N	.32	SN74163N	.99
SN7408N	.25	SN7479N	5.00	SN74164N	1.10
SN7409N	.25	SN7480N	.50	SN74165N	1.10
SN7410N	.18	SN7482N	.98	SN74166N	1.25
SN7411N	.30	SN7483N	.70	SN74167N	5.50
SN7412N	.33	SN7485N	.89	SN74170N	2.10
SN7413N	.45	SN7486N	.39	SN74172N	8.95
SN7414N	.70	SN7488N	3.50	SN74173N	1.50
SN7416N	.35	SN7489N	2.49	SN74174N	1.25
SN7417N	.35	SN7490N	.45	SN74175N	.99
SN7420N	.21	SN7491N	.75	SN74176N	.90
SN7421N	.33	SN7492N	.49	SN74177N	.90
SN7422N	.49	SN7493N	.49	SN74180N	.99
SN7423N	.37	SN7494N	.79	SN74181N	2.49
SN7425N	.29	SN7495N	.79	SN74182N	.95
SN7426N	.29	SN7496N	.89	SN74184N	1.95
SN7427N	.37	SN7497N	4.00	SN74185N	2.20
SN7429N	.42	SN74100N	1.00	SN74186N	15.00
SN7430N	.26	SN74107N	.39	SN74187N	6.00
SN7432N	.31	SN74121N	.39	SN74188N	3.95
SN7437N	.27	SN74122N	.39	SN74190N	1.19
SN7438N	.27	SN74123N	.50	SN74191N	1.25
SN7439N	.25	SN74125N	.60	SN74192N	.89
SN7440N	.15	SN74126N	.60	SN74193N	.89
SN7441N	.89	SN74132N	1.09	SN74194N	1.25
SN7442N	.59	SN74136N	.95	SN74195N	.75
SN7443N	.75	SN74141N	1.15	SN74196N	1.25
SN7444N	.75	SN74142N	4.00	SN74197N	.75
SN7445N	.75	SN74143N	4.50	SN74198N	1.75
SN7446N	.81	SN74144N	4.50	SN74199N	1.75
SN7447N	.69	SN74145N	1.15	SN74200N	5.59
SN7448N	.79	SN74147N	2.35	SN74279N	.90
SN7450N	.26	SN74148N	2.00	SN74251N	1.79
SN7451N	.27	SN74150N	1.00	SN74284N	6.00
SN7453N	.27	SN74151N	.79	SN74285N	6.00
SN7454N	.20	SN74153N	.89	SN74367N	.75

MANY OTHERS AVAILABLE ON REQUEST
20% Discount for 100 Combined 7400's

CMOS

CD4000	.25			74C04N	.75
CD4001	.25	CD4030	.65	74C10N	.65
CD4002	.25	CD4035	1.85	74C20N	.65
CD4006	2.50	CD4040	2.45	74C30N	.65
CD4007	.25	CD4042	1.90	74C42N	2.15
CD4009	.59	CD4044	1.50	74C73N	1.50
CD4010	.59	CD4046	2.51	74C74	1.15
CD4011	.25	CD4047	2.75	74C90N	3.00
CD4012	.25	CD4049	.79	74C95N	2.00
CD4013	.47	CD4050	.79	74C107N	1.25
CD4016	.56	CD4051	2.95	74C151	2.90
CD4017	1.35	CD4053	2.95	74C154	4.00
CD4019	.55	CD4060	3.25	74C157	2.15
CD4020	1.49	CD4066	1.75	74C160	3.25
CD4022	1.25	CD4069	.45	74C161	3.25
CD4023	.25	CD4071	.45	74C163	3.00
CD4024	1.50	CD4081	.45	74C164	3.25
CD4025	.25	CD4511	2.50	74C173	2.60
CD4026	$3.95	CD4518	2.50	74C193	2.75
CD4027	.69	MC14566	3.00	74C195	2.75
CD4028	1.65	74C00N	.39	MC4044	4.50
CD4029	2.90	74C02N	.55	MC14016	.56

LINEAR

LM300H	.80			LM1351N	1.65
LM301H	.35			LM1414N	1.75
LM301CN	.35	78MG	1.75	LM1458C	.65
LM302H	.75	LM370N	1.15	LM1496N	.95
LM304H	1.00	LM373N	3.25	LM1556V	1.85
LM305H	.95	LM377N	4.00	LM2111N	1.95
LM307CN	.35	LM380N	1.39	LM2901N	2.95
LM308H	1.00	LM380CN	1.05	LM3065N	.69
LM308CN	1.00	LM381N	1.79	LM3900N	.55
LM309H	1.10	LM382N	1.79	LM3905N	.60
LM309K	.99	NE501K	8.00	LM3909	1.25
LM310CN	1.15	NE510A	6.00	LM5556N	1.85
LM311H	.90	NE531H	3.00	MC5558V	1.00
LM311N	.90	NE536T	6.00	LM7525N	.90
LM318CN	1.50	NE540L	6.00	LM7535N	1.25
LM319N	1.30	NE550N	.79	8038B	4.95
LM320K-5	1.35	NE555V	.39	LM75450	.49
LM320K-5.2	1.35	NE560B	5.00	75451CN	.39
LM320K-12	1.35	NE561B	5.00	75452CN	.39
LM320K-15	1.35	NE562B	5.00	75453CN	.39
LM320T-5	1.75	NE565H	1.25	75454CN	.39
LM320T-5.2	1.75	NE565N	1.75	75491CN	.79
LM320T-8	1.75	NE566CN	1.25	75492CN	.89
LM320T-12	1.75	NE567H	1.95	75494CN	.89
LM320T-15	1.75	NE567V	1.50	**RCA LINEAR**	
LM320T-18	1.75	LM703CN	.45	CA3013	2.15
LM320T-24	1.75	LM709H	.29	CA3023	2.56
LM323K-5	9.95	LM709N	.29	CA3035	2.48
LM324N	1.80	LM710N	.79	CA3039	1.35
LM339N	1.70	LM711N	.39	CA3046	1.30
LM340K-5	1.95	LM723H	.55	CA3059	3.25
LM340K-6	1.95	LM723H	.55	CA3060	3.25
LM340K-8	1.95	LM733N	1.00	CA3080	.85
LM340K-12	1.95	LM739N	1.00	CA3081	2.00
LM340K-15	1.95	LM741CH	.35	CA3082	2.00
LM340K-18	1.95	LM741CN	.35	CA3083	1.60
LM340K-24	1.95	LM741-14N	.39	CA3086	.85
LM340T-5	1.75	LM747H	.79	CA3089	3.75
LM340T-6	1.75	LM747N	.79	CA3091	10.20
LM340T-8	1.75	LM748H	.39	CA3102	2.95
LM340T-12	1.75	LM748N	.39	CA3123	2.15
LM340T-15	1.75	LM1303N	.90	CA3130	1.39
LM340T-18	1.75	LM1304N	1.19	CA3140	1.25
LM340T-24	1.75	LM1305N	1.40	CA3600	1.75
LM350N	1.00	LM1307N	.85	RC4194	5.95
LM351CN	.65	LM1310N	2.95	RC4195	3.25

74LS00 TTL

74LS00	.29			74LS139	1.95
74LS02	.29			74LS151	1.55
74LS03	.29	74LS74	.49	74LS153	1.89
74LS04	.35	74LS75	.69	74LS157	1.55
74LS05	.35	74LS76	.49	74LS162	2.25
74LS08	.29	74LS83	1.75	74LS163	2.25
74LS10	.29	74LS85	2.49	74LS164	1.95
74LS13	.69	74LS86	.49	74LS175	1.95
74LS14	1.75	74LS90	1.25	74LS181	3.69
74LS20	.29	74LS92	1.25	74LS190	2.85
74LS26	.39	74LS93	1.25	74LS191	2.85
74LS27	.39	74LS95	1.95	74LS192	2.85
74LS28	.39	74LS96	1.89	74LS193	2.85
74LS30	.29	74LS107	.59	74LS194	1.89
74LS32	.39	74LS109	.59	74LS195	1.89
74LS40	.39	74LS112	.59	74LS257	1.75
74LS51	.29	74LS132	1.25	74LS260	.55
74LS55	.29	74LS136	.59	74LS279	.79
74LS73	.49	74LS138	1.89	74LS670	3.95

CLOCK CHIPS

MM5309	6 Digit, BCD Outputs, Reset PIN.	$9.95
MM5311	6 Digit, BCD Outputs, 12 or 24 Hour	4.95
MM5312	4 Digit, BCD Outputs, 1 PPS Output	4.95
MM5314	6 Digit, 12 or 24 Hour, 50 or 60 Hz	4.95
MM5316	4 Digit, Alarm, 1 PPS Output	6.95
MM5318	Video Clock Chip, For Use With (MM5841 - $9.95)	9.95
CT7001	6 Digit, Calendar, Alarm, 12 or 24 Hour	5.95

THE PROM SETTER

WRITE and READ EPROM

1702A and 2708

- Plugs Directly into your ALTAIR/IMSAI Computer
- Includes Main Module Board and External EPROM Socket Unit
- The EPROM Socket Unit is connected to the Computer through a 25 Pin Connector
- Programming is accomplished by the Computer
- Just Read in the Program to be Written on the EPROM into your Processor and let the Computer do the rest.
- Use Socket Unit to Read EPROM's Contents into your Computer
- Software included
- No External Power Supplies, Your Computer does it all
- Programs and Reads Both 1702A and 2708 EPROMS
- Doubles as an Eight Bit Parallel I/O
- Manual included *Delivery Less than 90 days*

INTRODUCTRY OFFER

THE PROM SETTER
Kit **$210** ASSEMBLED **$375**

PROM SETTER I
above unit with 2716 adapter
Kit **$260** Assembled **$425**

WIRE WRAP CENTER

HOBBY-WRAP TOOL-BW-630



$34.95 (batteries not included)

- Battery Operated (Size C)
- Weighs ONLY 11 Ounces
- Wraps 30 AWG Wire onto Standard DIP Sockets (.025 inch)
- Complete with built-in bit and sleeve

WIRE-WRAP KIT — WK-2-W

WRAP • STRIP • UNWRAP



- Tool for 30 AWG Wire
- Roll of 50 Ft. White or Blue 30 AWG Wire
- 50 pcs. each 1", 2", 3" & 4" lengths — pre-stripped wire.

$11.95



WIRE WRAP TOOL WSU-30

WRAP • STRIP • UNWRAP - $5.95

WIRE WRAP WIRE — 30 AWG

50 ft. $1.95 1000 ft. $15.00

SPECIFY COLOR — White - Yellow - Red - Green - Blue - Black

WIRE DISPENSER — WD-30

- 50 ft. roll 30 AWG KYNAR wire wrap wire **$3.45 ea.**
- Cuts wire to desired length
- Strips 1" of insulation **Specify** — Blue-Yellow-White-Red

LIQUID CRYSTAL DIGITAL CLOCK-CALENDAR



For Auto, Home, Office
Small in size (2x2½x½)
Push button for seconds release for date.
Clocks mount anywhere with either 3M double-sided tape or VELCRO, included.
2 MODELS AVAILABLE:
LCD-101, portable model runs on self-contained batteries for better than a year.
LCD-102, runs on 12 Volt system and is back-lighted.

LCD-101 or LCD-102 your choice........................ **$33.95** ea.

Clear desk stand for........................ **$2.00**

MA1003, 12V DC CLOCK MODULE



$17.95

Built in X'TAL controlled time base. Protected against automotive volt transients. Automatic brightness control with 0.3" green color display. Display turnoff with ignition "OFF"

JADE Computer Products

5351 WEST 144th STREET
LAWNDALE, CALIFORNIA 90260
(213) 679-3313

RETAIL STORE HOURS M-F 9-7 SAT. 9-5

Discounts available at OEM quantities. Add $1.25 for shipping. California residents add 6% sales tax.

CATALOG FREE WITH $10.00 ORDER

Circle 66 on inquiry card.

MICROCOMPUTER

MICROPROCESSOR's

F-8	19.95
Z-80	25.00
Z-80A	35.00
CDP1802CD	24.95
AM2901	22.95
6502	12.95
6800	19.95
8008-1	8.75
8080A	15.95
TMS-9900TL	89.95

6800 SUPPORT

6810P	4.95
6820P	8.00
6828P	11.25
6834P	16.95
6850P	9.95
6852P	11.95
6860P	14.95
6862P	17.95
6880P	2.70

Z80 SUPPORT DEVICES

3881	12.95
3882	12.95

F-8 SUPPORT DEVICES

3851	14.95
3853	14.95

DYNAMIC RAMS

1103	1.50
2104	4.50
2107A	3.75
2107B	4.50
2107B-4	4.00
TMS4050	4.50
TMS4060	4.50
4096	4.50
4116	42.00
MM5270	5.00
MM5280	6.00
MCM6605	6.00

CHARACTER GENERATORS

2513 UP	6.75
2513 DOWN	6.75
2513 UP (5v)	9.95
2513 DOWN(5v)	10.95
MCM6571	10.80
MCM6571A	10.80
MCM6572	10.80
MCM6574	14.75
MCM6575	14.75

PROM'S

1702A	5.00
2704	15.00
2708	20.00
2716	38.00
3601	4.50
5203AQ	4.00
5204AQ	6.00
6834	16.95
6834-1	14.95
82S23B	4.00
82S129B	4.25
8223B	2.70

8080A SUPPORT DEVICES

8212	3.95
8214	9.95
8216	4.50
8224	4.95
8228	8.75
8238	8.00
8251	12.00
8253	28.00
8255	12.00
8257	25.00
8259	25.00

STATIC RAMS	1-24	25-99	100
21L02 (450)	1.50	1.40	1.25
21L02 (250)	1.95	1.80	1.50
21L11	4.25	4.10	3.95
1101A	1.49	1.29	1.10
2101-1	2.95	2.75	2.60
2102	1.25	1.15	1.00
2102-1	1.50	1.30	1.15
2111-1	4.00	3.50	3.25
2112-1	3.00	2.80	2.69
2114	17.95	16.95	16.50
31L01	2.50	2.35	2.00
3107	3.95	3.70	3.25
4200A	12.95	12.50	11.95
4804/2114	17.95	16.95	16.50
5101C-E	11.95	11.25	10.25
74C89	3.25	3.05	2.85
74S201	4.50	4.30	4.25
7489	2.25	2.10	1.90
8599	1.88	1.75	1.60
9102BPC	1.65	1.45	1.30

KEYBOARD CHIPS

AY5-2376	13.95
AY5-3600	13.95

UART'S

AY5-1013A	5.50
AY5-1014A	8.95
TR-1602A	5.50
TMS-6011	6.95
IM-6402	10.80
IM-6403	10.80

FLOPPY DISC CONTROLLER

1771B	55.95
1771B-01	59.95

SHIFT REGISTERS STATIC

2518B	3.95
2533V	2.00
TMS3002	1.00
TMS3112	3.95
MM5058	2.00

KIM

KIM-1	245.00
6502	12.95
6520	9.00
6522	9.25
6530-002	15.95
6530-003	15.95
6530-004	15.95
6530-005	15.95

USRT

S-2350	10.95
WD1671B	29.95

WAVEFORM GENERATOR

8038	4.00
MC4024	2.50
566	1.75

MISC. OTHER COMPONENTS

NH0025CN	1.70	P-3408A	5.00
NH0026CN	2.50	P-4201	4.95
N8T20	3.50	MM-5320	7.50
N8T26	2.45	MM-5369	1.90
74367	.90	DM-8130	2.90
DM8098	.90	DM8131	2.75
1488	1.95	DM-8831	2.50
1489	1.95	DM-8833	2.50
D-3207A	2.00	DM-8835	2.50
C-3404	3.95	SN74LS367	.90
		SN74LS368	.90

TV Game Chip
TMS1955NL
Now Only $10.95

PerSci DISK AND CONTROLLER

Use the PerSci Disk and Controller now with the Info 2000 Adapter for the S-100 Bus.

INFO 2000 "SPECIAL" (includes Model 277 Dual Drive, Model 1070 Controller, Case with power supply and fan, and cable)	$2,150
Model 277 Dual Diskette Drive	$1,130
Model 1070 Controller	$740
Slimline case with power supply and fan	$280
Adapter for the S-100 Bus (Kit)	$120

JADE PARALLEL/SERIAL INTERFACE KIT $124.95 KIT

JADE VIDEO INTERFACE KIT $89.95 KIT

8K STATIC RAM BOARD

ASSEMBLED & TESTED

250ns.	$199.95
350ns.	$189.95
450ns.	$169.95

* WILL WORK WITH NO FRONT PANEL
* FULL DOCUMENTATION
* FULLY BUFFERED
* S100 DESIGN
* ADEQUATELY BYPASSED
* LOW POWER SCHOTTKY SUPPORT IC'S

KIT

250ns.	$169.95
350ns.	$139.95
450ns.	$129.95

IMSAI/ALTAIR **S-100** COMPATIBLE

JADE Z80 KIT

—with PROVISIONS for ONBOARD 2708 and POWER ON JUMP

$135.00 EA. (2MHZ)

$149.95 EA. (4MHZ)

BARE BOARD **$35.00**

IMSAI/ALTAIR **S-100** COMPATIBLE

JADE 8080A KIT

– WITH EIGHT LEVEL VECTOR INTERRUPT

$110.00 KIT

BARE BOARD **$35.00**

JADE
Computer Products
5351 WEST 144th STREET
LAWNDALE, CALIFORNIA 90260
(213) 679-3313

RETAIL STORE HOURS M-F 9-7 SAT. 9-5

Discounts available at OEM quantities. Add $1.25 for shipping. California residents add 6% sales tax.

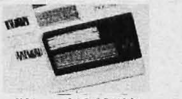
CATALOG FREE WITH $10.00 ORDER

CRYSTALS

THESE FREQUENCIES ONLY

Part #	Frequency	Case/Style	Price
CY1A	1.000 MHz	HC33/U	$5.95
CY2A	2.000 MHz	HC33/U	$5.95
CY2.01	2.010 MHz	HC33/U	$1.95
CY3A	4.000 MHz	HC18/U	$4.95
CY7A	5.000 MHz	HC18/U	$4.95
CY12A	10.000 MHz	HC18/U	$4.95
CY14A	14.31818 MHz	HC18U	$4.95
CY19A	18.000 MHz	HC18/U	$4.95
CY22A	20.000 MHz	HC18/U	$4.95
CY30B	32.000 MHz	HC18/U	$4.95

BREAD BOARD JUMPER WIRE KIT



JK1 ....$10.00 / kit

Each kit contains 350 wires cut to 14 different lengths from 0.1" to 5.0'.

Each wire is stripped and the leads are bent 90° for easy insertion.

Wire length is classified by color coding.

All wire is solid tinned 22 gauge with PVC insulation.

The wires come packed in a convenient plastic box.

CONNECTORS

PRINTED CIRCUIT EDGE-CARD

.156 Spacing-Tin-Double Read-Out

Bifurcated Contacts — Fits .054 to .070 P.C. Cards

15/30	PINS (Solder Eyelet)	$1.95
18/36	PINS (Solder Eyelet)	$2.49
22/44	PINS (Solder Eyelet)	$2.95
50/100	PINS (Wire Wrap)	$6.95
50/100A (.100 Spacing)	PINS (Wire Wrap)	$6.95

25 PIN-D SUBMINATURE

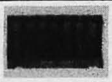
DB25P	PLUG	$3.25
DB25S	SOCKET	$4.95

HEAT SINKS

205-CB 680-.75A

205-CB	Beryllium Copper w-black finish for T0-5	$ .25
291-.36H	Aluminum for T0-220 Transistors & Regulators	$ .25
680-.75A	Black Anodized Aluminum for T0-3	$1.60
Dude 4	Black Anodized Aluminum — predrilled mounting holes for T0-3 — 4¾ x 1¼ x 2"	$1.75

DIP SWITCHES SPST Slide Action



#206-4 (8 pin dip) 4 switch unit $1.75 ea.
#206-7 (14 pin dip) 7 switch unit $1.95 ea.
#206-8 (16 pin dip) 8 switch unit $2.25 ea.

Etching Kits (cannot be shipped via air)

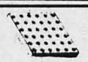
32 X A-1	P.C. Etch Materials Kit enough for 5 circuit boards	$29.95 ea.
27 X A-1	Etched Circuit Kit Complete kit — only add water	$ 9.95 ea.

Plugboards

3662	6.5 X 4.5 X 1/16 Epoxy glass P-Pattern-44 P.C. Tabs-spaced .156"	$ 6.95 ea.
22/44	Mating connector for plugboard — 22 pin double readout	$ 2.95 ea.
8800V	Universal Microcomputer/Processor plugboard — Epoxy Glass — complete with heatsink and mounting hardware 5.313 X 10 X 1/16 copper clad	$19.95 ea.

Vector

1/16 VECTOR BOARD



	0.1" Hole Spacing Part No.	P-Pattern L	W	Price 1-9	10 up
PHENOLIC	64P44 062XXXP	4.50	6.50	1.72	1.54
	169P44 062XXXP	4.50	17.00	3.69	3.32
EPOXY	64P44 062WE	4.50	6.50	2.07	1.86
GLASS	84P44 062WE	4.50	8.50	2.56	2.31
	169P44 062WE	4.50	17.00	5.04	4.53
	169P84 062WE	8.50	17.00	9.23	8.26
EPOXY GLASS COPPER CLAD	169P44 062WEC1	4.50	17.00	6.80	6.12

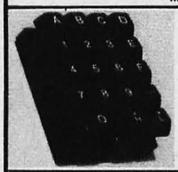
Vector

SLIT-N-WRAP WIRE WRAP TOOL

- Slits and opens insulation exposing bare wire
- No pre-cutting or pre-stripping.
- Comes complete with two - 100 ft spools #28 AWG wire

Model P180 $24.50

HEXADECIMAL ENCODER 19-KEY PAD



- 1 - 0
- ABCDEF
- Shift Key
- 2 Optional Keys

$10.95 each

63 KEY KEYBOARD $29.95



This keyboard features 63 unencoded SPST keys, unattached to any kind of P.C.B. A very solid molded plastic 13" x 4" base suits most applications.

HD0165	Encoder Chip (encodes 16 Keys)	$7.95 ea.
AY-5-2376	Encoder Chip (encodes 88 Keys)	$14.95 ea.

TOOLS

A97MS	— Diagonal Cutter - 4" semi-flush cut	$7.95 ea.
A11DMS	— Chain Nose Pliers - 4¾" long	7.50 ea.
T-6	— Wire Stripper - #16 to #26 gauge	3.75 ea.
55B	— Wire Stripper - #10 to #20 gauge	2.50 ea.
CS-8	— Cutter-Crimper Tool - 8¼" long	7.95 ea.
Nibbling Tool	— Cuts, Trims or Notches Metal up to #18 gauge	6.95 ea.
Nibbling Tool Replacement Punch		3.75 ea.

PERMACEL® P-29 PLUS Electrical Tape - All Weather
• ¾" wide x 66 ft. • Black vinyl
1-9 Rolls $.79 each 10-up Rolls $6.95/10 roll package

MICROPROCESSOR COMPONENTS

8080A	CPU	$16.00	CDP1802	CPU	$19.95
8212	8 Bit Input/Output	4.95	MC6800	8 Bit MPU	24.95
8214	Priority Interrupt Control	15.95	MC6820	Periph. Interface Adapter	15.00
8216	Bi-Directional Bus Driver	6.95	MC6810AP1	128 x 8 Static RAM	6.00
8224	Clock Generator/Driver	9.95	MC6830L8	1024 x 8 Bit ROM	15.00
8228	System Controller Bus Driver	10.95	Z80	CPU	29.95

CPU'S

8080A	Super 8008	16.00
2650	8 BIT MPU	26.50
P8085	CPU	29.95

SR'S

2504	1024 Dynamic	$ 3.95
2518	Hex 32 BIT	7.00
2519	Hex 40 BIT	4.00
2522	Dual 132 Bit SSR	2.95
2524	512 Dynamic	1.95
2525	1024 Dynamic	3.00
2527	Dual 256 BIT	3.95
2529	Dual 512 BIT	4.00
2532	Quad 80 BIT	3.95
2533	1024 Static	5.95
3341	Fifo	6.95
74LS670	16 x 4 Reg	3.95

UART'S

AY-5-1013	30K Baud	$5.95

ROM'S

2513(2140)	Char. Gen.-upper case	$ 9.95
2513(3021)	Char. Gen.-lower case	9.95
2516	Char. Gen.	10.95
MM5230	2048 BIT (512 x 4 on 256 x 8)	1.95

RAM'S

1101	256 x 1	Static	$ 1.49
2101	256 x 4	Static	5.95
2102	1024 x 1	Static	1.75
2107/5280	4096 x 1	Dynamic	4.95
2111	256 x 4	Static	6.95
7489	16 x 4	Static	2.49
8101	256 x 4	Static	5.95
8111	256 x 4	Static	6.95
8599	16 x 4	Static	3.49
21L02/91L02	1024 x 1	Static	2.25
74200	256 x 1	Static	6.95
93421	256 x 1	Static	2.95
MM5262	2K x 1	Dynamic	2 for 1.00
UPD414(2104)	4K	Dynamic 16 Pin	5.95

PROMS

1702A	2048	Famos	$ 9.95
5203	2048	Famos	14.95
82S23	32 x 8	Open C	5.00
82S123	32 x 8	Tristate	5.00
74S287	1024	Static	7.95
3601	256 x 4	Fast	3.95
2708	8K	Eprom	19.95
2716	16K	Eprom	34.95
6301-1	1024	Tri-State Bipolar	3.49
6330-1	256	Open Collector Bipolar	2.95

SPECIAL REQUESTED ITEMS

FCM3817	$5.00	11C90	19.95	MCM6575	17.50	9368	3.95
AY-3-8500-1	8.95	4N33	3.95	ICM7045	24.95	LD110/111	25.00/set
AY-5-9100	17.50	8T20	7.50	ICM7207	7.50	95H90	11.95
AY-5-9200	14.95	8T97	2.00	ICM7208	22.00	MC3061P	3.50
AY-5-9500	4.95	HD0165	7.95	ICM7209	7.50	MC4016 (74416)	7.50
AY-5-2376	14.95	MCM6571	17.50	MK50240	17.50	MC1408L7	8.95
9374	1.95	MCM6574	17.50	DS0026CH	3.75	MC1408L8	9.95
82S115	25.00	7205	19.95	TIL308	10.50		

CLOCK CHIPS

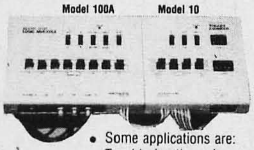
MM5309	$9.95
MM5311	4.95
MM5312	4.95
MM5314	4.95
MM5316	6.95
MM5318	9.95
MM5369	2.95
MM5841	9.95
CT7001	5.95

PARATRONICS

Featured on February's Front Cover of Popular Electronics

Logic Analyzer Kit Model 100A

$229.00/kit



- Analyzes any type of digital system
- Checks data rates in excess of 8 million words per second
- Trouble shoot TTL, CMOS, DTL, RTL, Schottky and MOS families
- Displays 16 logic states up to 8 digits wide
- See ones and zeros displayed on your CRT, octal or hexadecimal format
- Tests circuits under actual operating conditions
- Easy to assemble — comes with step-by-step construction manual which includes 80 pages on logic analyzer operation.

- Some applications are:
  - — Troubleshooting microprocessor address, instruction, and data flow
  - — Examine contents of ROMS
  - — Tracing operation of control logic
  - — Checking counter and shift register operation
  - — Monitoring I/O sequences
  - — Verifying proper system operations during testing

PARATRONICS TRIGGER EXPANDER - Model 10

Adds 16 additional bits. Provides digital delay and qualification of input clock and 24-bit trigger word. — Connects direct to Model 100A for integrated unit). **Model 10 — $229.00** Baseplate — $6.95

B+K PRECISION 3½-Digit Portable DMM



- Overload Protected
- .3" high LED Display
- Battery or AC operation
- Auto Zeroing
- 1mv, 1Va, 0.1 ohm resolution
- Overange reading
- 10 meg input impendence
- DC Accuracy 1% typical

Ranges: DC Voltage ±0-1000V/
AC Voltage: 0-1000V
Freq. Response: 50-400 HZ
DC/AC Current: 0-100mA
Resistance: 0-10 meg ohm
Size: 6.4" x 4.4" x 2"

Model 2800 $99.95

Comes with test leads, operating manual and spare fuse.

Accessories:

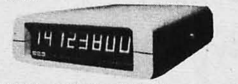
AC Adapter BC-28	8.00
Rechargeable Batteries BP-26	20.00
Carrying Case LC-28	6.00

CSC — 100 MHz 8-Digit Counter

CONTINENTAL SPECIALTIES CORPORATION

- 20 Hz-100 MHz Range
- .6" LED Display
- Crystal-controlled timebase
- Fully Automatic
- Portable — completely self-contained
- Size — 1.75" x 7.38" x 5.63"
- Four power souces, i.e. batteries, 110 or 220V with charger 12V with auto lighter adapter and external 7.2-10V power supply

MAX-100 $134.95



ACCESSORIES FOR MAX 100:

Mobile Charger Eliminator use power from car battery — Model 100 — CLA $3.95

Charger/Eliminator use 110 V AC — Model 100 — CAI $9.95

CONTINENTAL SPECIALTIES

PROTO BOARD 6

$15.95
(6" long X 4" wide)



Other CS Proto Boards

PB100 - 4.5" x 6"	$ 19.95
PB101 - 5.8" x 4.5"	29.95
PB102 - 7" x 4.5"	39.95
PB103 - 9" x 6"	59.95
PB104 - 9.5" x 8"	79.95
PB203 - 9.75 x 6½ x 2¾	80.00
PB203A - 9.75 x 6½ x 2¾ (includes power supply)	129.95

PROTO CLIPS

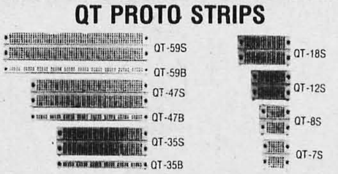
14 PIN	$4.50
16 PIN	4.75
24 PIN	8.50
40 PIN	13.75

DESIGN MATES

DM1 - Circuit Designer $69.95
DM2 - Function Generator $74.95
DM3 - RC Bridge $74.95

LOGIC MONITOR for DTL, HTL, TTL or CMOS Devices **$84.95**

QT PROTO STRIPS



QT type	#holes	price
QT-59S	590	12.50
QT-59B	bus strip	2.50
QT-47S	470	10.00
QT-47B	bus strip	2.25
QT-35S	350	8.50
QT-35B	bus strip	2.00
QT-18S	180	4.75
QT-12S	120	3.75
QT-8S	80	3.25
QT-7S	70	3.00

Experimentor 300	$ 9.95
Experimentor 600	$10.95

$5.00 Minimum Order — U.S. Funds Only
California Residents — Add 6% Sales Tax

Spec Sheets - 25¢ — Send 35¢ Stamp for 1978 Catalog
Dealer Information Available

NOW OPEN SATURDAYS

James ELECTRONICS

1978 CATALOG NOW AVAILABLE

1021-A HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097
All Advertised Prices Good Thru December

Timeband

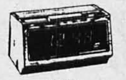
A Trademark of Fairchild Camera and Instrument Corporation

Digital Alarm Clocks



C-500 - Ivory Case
C-500B - Ebony Case
$16.95

- 24-hour alarm
- Doze Button
- 100% Solid State
- Large Red Led Display (.8" high)
- AM/PM Indicator
- Seconds display at touch of button
- Power failure indicator
- One year factory warranty



C-8211
Woodgrain Case
$19.95

Timeband™ by FAIRCHILD — LAMP CLOCK



- 24 hour alarm
- Doze button
- Alarm-on indicator
- .8" high Red LED Display
- AM/PM Indicator
- High intensity lamp
- Lamp shuts off when collapsed

Model C-590 (Ivory) $29.95

ERC AM/FM 8-Track Stereo Receiver With BSR Changer



- PLL System
- BSR Record Changer
- Slide Controls
- Automatic AFC Control
- 4 Speaker Output
- Walnut finish vinyl covered wood veneer with smoke dust cover

Size: 20"W x 9⅜"H x 15¼"D

Model 8365 $149.95

DIGITAL STOPWATCH

- Bright 6 Digit LED Display
- Times to 59 minutes 59.59 seconds
- Crystal Controlled Time Base
- Three Stopwatches in One
  Times Single Event — Split & Taylor
- Size 4.5" x 2.15" x .90" (4½ ounces)
- Uses 3 Penlite Cells

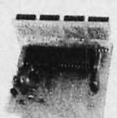


Kit — $39.95
Assembled — $49.95
Heavy Duty Carry Case $5.95

Stop Watch Chip Only (7205) $19.95

IMC 3½ DIGIT DPM KIT

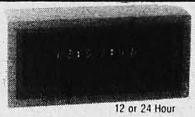




- New Bipolar Unit
- Auto Zeroing
- .5" LED
- Auto Polarity
- Low Power
- Single IC Unit

Model KB500 DPM Kit $49.00
Model 311D-5C-5V Power Kit $17.50

JE700 CLOCK



12 or 24 Hour

The JE700 is a low cost digital clock, but is a very high quality unit. The unit features a simulated walnut case with dimensions of 6" x 2½" x 1". It utilizes a MAN72 high brightness readout, and the MM5314 clock chip.

115 VAC *KIT ONLY* **$16.95**

JOYSTICK

These joysticks feature four 100K potentiometers, that vary resistance proportional to the angle of the stick. Study metal construction with plastic components only at the movable joint. Perfect for electronic games and instrumentation.

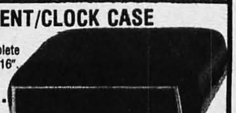


100K	$5.95
5K	$4.95

INSTRUMENT/CLOCK CASE

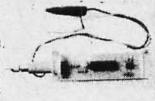
Injection molded unit. Complete with red bezal. 4½" x 4" x 1-9/16".

$3.95 ea.



JE803 PROBE

The Logic Probe is a unit which is for the most part, indespensible in trouble shooting logic families TTL, DTL, RTL, CMOS. It derives the power it needs to operate directly off of the circuit under test, drawing a scant 10 mA max. It uses a MAN3 readout to indicate any of the following states by these symbols (H) - 1 (LOW) - o (PULSE) - P. The Probe can detect high frequency pulses to 45 MHz. It can't be used at MOS levels or circuit damage will result.



$9.95 Per Kit
printed circuit board

T²L 5V 1A Supply



This is a standard TTL power supply using the well known LM309K regulator IC to provide a solid 1 AMP of current at 5 volts. We try to make things easy for you by providing everything you need in one package, including the hardware for only: **$9.95 Per Kit**

Circle 67 on inquiry card.

7400N TTL

SN7400N	.16	SN7472N	.39	SN74160N	1.25
SN7401N	.18	SN7473N	.39	SN74161N	.99
SN7402N	.20	SN7474N	.35	SN74162N	1.95
SN7403N	.20	SN7475N	.50	SN74163N	.99
SN7404N	.20	SN7476N	.35	SN74164N	.99
SN7405N	.20	SN7479N	5.00	SN74165N	.99
SN7406N	.35	SN7480N	.50	SN74166N	1.25
SN7407N	.35	SN7482N	.99	SN74167N	3.25
SN7408N	.20	SN7483N	.70	SN74170N	2.10
SN7409N	.25	SN7485N	.89	SN74172N	6.00
SN7410N	.20	SN7486N	.39	SN74173N	1.50
SN7411N	.30	SN7488N	3.50	SN74174N	1.25
SN7412N	.35	SN7489N	2.49	SN74175N	.99
SN7413N	.69	SN7490N	.45	SN74176N	.79
SN7414N	.70	SN7491N	.75	SN74177N	.79
SN7416N	.35	SN7492N	.49	SN74179N	2.49
SN7417N	.35	SN7493N	.49	SN74180N	.99
SN7420N	.20	SN7494N	.79	SN74181N	2.49
SN7421N	.39	SN7495N	.79	SN74182N	.95
SN7422N	.49	SN7496N	.79	SN74184N	1.95
SN7423N	.37	SN7497N	3.00	SN74185N	1.95
SN7425N	.29	SN74100N	1.25	SN74186N	15.00
SN7426N	.29	SN74107N	.39	SN74187N	6.00
SN7427N	.37	SN74109N	.95	SN74188N	3.95
SN7429N	.42	SN74116N	1.95	SN74190N	1.79
SN7430N	.25	SN74121N	.39	SN74191N	1.25
SN7432N	.25	SN74122N	.39	SN74192N	.89
SN7437N	.35	SN74123N	.50	SN74193N	.89
SN7438N	.35	SN74125N	.60	SN74194N	1.25
SN7439N	.25	SN74126N	.60	SN74195N	.75
SN7440N	.21	SN74132N	1.25	SN74196N	1.00
SN7441N	.89	SN74136N	.95	SN74197N	1.00
SN7442N	.69	SN74141N	1.15	SN74198N	1.75
SN7443N	.75	SN74142N	2.95	SN74199N	1.75
SN7444N	.75	SN74143N	3.25	SN74200N	5.59
SN7445N	.75	SN74144N	3.00	SN74251N	1.79
SN7446N	.89	SN74145N	1.15	SN74279N	.90
SN7447N	.69	SN74147N	2.35	SN74283N	2.25
SN7448N	.89	SN74148N	2.00	SN74284N	6.00
SN7450N	.25	SN74150N	1.25	SN74285N	6.00
SN7451N	.25	SN74151N	.79	SN74365N	.75
SN7453N	.25	SN74153N	.89	SN74366N	.75
SN7454N	.25	SN74154N	1.25	SN74367N	.75
SN7459A	.25	SN74155N	.89	SN74368N	.75
SN7460N	.25	SN74156N	.89	SN74390N	2.25
SN7470N	.45	SN74157N	.89	SN74393N	2.25

20% Discount for 100 Combined 7400's

CMOS

CD4000	.23	CD4044	.89	MC14562	14.50
CD4001	.23	CD4046	1.79	CD4566	2.25
CD4002	.23	CD4047	2.50	MC14583	3.50
CD4006	1.19	CD4048	1.35		
CD4007	.25	CD4049	.49	**74C00 Series**	
CD4009	.49	CD4050	.49	74C00	.39
CD4010	.49	CD4051	1.19	74C02	.55
CD4011	.23	CD4053	1.19	74C04	.75
CD4012	.25	CD4056	1.49	74C10	.65
CD4013	.39	CD4059	9.95	74C14	3.00
CD4014	1.39	CD4060	1.49	74C20	.65
CD4015	1.19	CD4066	.79	74C30	.65
CD4016	.49	CD4068	.39	74C42	2.15
CD4017	1.19	CD4069	.45	74C73	1.50
CD4018	.99	CD4070	.55	74C74	1.15
CD4019	.49	CD4071	.23	74C89	4.00
CD4020	1.19	CD4072	.49	74C90	3.00
CD4021	1.39	CD4076	1.39	74C93	2.00
CD4022	1.19	CD4081	.23	74C95	2.00
CD4023	.23	CD4082	.23	74C107	1.25
CD4024	.79	CD4098	2.49	74C151	2.90
CD4025	.23	MC14409	14.95	74C154	3.00
CD4026	2.25	MC14410	14.95	74C157	2.15
CD4027	.69	MC14419	4.95	74C160	3.25
CD4028	.89	MC14506	.75	74C161	3.25
CD4029	1.19	MC14507	.99	74C163	3.00
CD4030	.49	CD4508	3.95	74C164	3.25
CD4035	.99	CD4510	1.39	74C173	2.60
CD4040	1.19	CD4511	1.29	74C193	2.75
CD4041	1.25	CD4515	2.95	74C195	2.75
CD4042	.99	CD4518	1.29	80C95	1.50
CD4043	.89	CD4520	1.29	80C97	1.50

LINEAR

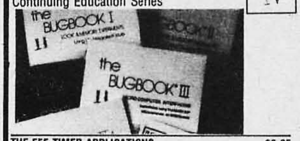
LM300H	.80			LM739N	1.19
LM301H	.35	LM340T-8	1.25	LM741CH	.35
LM301CN	.35	LM340T-12	1.25	LM741CN	.35
LM302H	.75	LM340T-15	1.25	LM741-14N	.39
LM304H	1.00	LM340T-18	1.25	LM747H	.79
LM305H	.60	LM340T-24	1.25	LM747N	.79
LM307CN	.35	LM350N	1.00	LM748H	.39
LM308H	1.00	LM351CN	.65	LM748N	.39
LM308CN	1.00	78MG	1.75	LM1303N	.90
LM309H	1.10	LM370N	1.15	LM1304N	1.19
LM309K	1.25	LM373N	3.25	LM1305N	1.40
LM310CN	1.15	LM377N	4.00	LM1307N	.85
LM311H	.90	LM380N	1.25	LM1310N	2.95
LM311N	.90	LM380CN	.99	LM1351N	1.65
LM317K	6.50	LM381N	1.79	LM1414N	1.75
LM318CN	1.50	LM382N	1.79	LM1458CN	.59
LM319N	1.30	NE501K	8.00	LM1496N	.95
LM320K-5	1.35	NE510A	6.00	LM1556V	1.75
LM320K-5.2	1.35	NE529A	4.95	LM2111N	1.95
LM320K-12	1.35	NE531H	3.00	LM2901N	2.95
LM320K-15	1.35	NE536T	6.00	LM3053	1.50
LM320T-5	1.25	NE540L	6.00	LM3065N	.69
LM320T-5.2	1.25	NE550N	1.30	LM3900N(3401)	.49
LM320T-8	1.25	NE555V	.39	LM3905N	.89
LM320T-12	1.25	NE560B	5.00	LM3909	1.25
LM320T-15	1.25	NE561B	5.00	LM5556N	1.85
LM320T-18	1.25	NE562B	5.00	MC5558V	1.00
LM320T-24	1.25	NE565H	1.75	LM7525N	.90
LM323K-5	5.95	NE565N	1.25	LM7534N	.75
LM324N	1.80	NE566CN	1.75	8038B	4.95
LM339N	.99	NE567H	1.95	LM75450	.49
LM340K-5	1.35	NE567V	1.49	75451CN	.39
LM340K-6	1.35	LM703CN	.45	75452CN	.39
LM340K-8	1.35	LM709H	.29	75453CN	.39
LM340K-12	1.35	LM709N	.29	75454CN	.39
LM340K-15	1.35	LM710N	.79	75491CN	.79
LM340K-18	1.35	LM711N	.39	75492CN	.89
LM340K-24	1.35	LM723H	.55	75494CN	.89
LM340T-5	1.25	LM723N	.55	RC4194	5.95
LM340T-6	1.25	LM733N	1.00	RC4195	3.25

74LS00 TTL

74LS00	.29			74LS155	1.25
74LS02	.29			74LS157	1.50
74LS03	.29	74LS75	.69	74LS160	1.95
74LS04	.35	74LS76	.49	74LS161	1.95
74LS05	.35	74LS83	1.75	74LS162	1.95
74LS08	.29	74LS85	2.49	74LS163	1.95
74LS10	.29	74LS86	.49	74LS164	1.95
74LS13	.69	74LS90	.89	74LS175	1.95
74LS14	1.75	74LS92	.89	74LS181	3.69
74LS20	.29	74LS93	.89	74LS190	2.49
74LS26	.39	74LS95	1.50	74LS191	2.49
74LS27	.39	74LS96	1.89	74LS192	2.49
74LS28	.39	74LS107	.59	74LS193	2.49
74LS30	.29	74LS109	.59	74LS194	1.89
74LS32	.39	74LS112	.59	74LS195	1.89
74LS40	.39	74LS123	1.25	74LS253	1.75
74LS42	1.25	74LS132	1.25	74LS257	1.75
74LS47	1.25	74LS136	.59	74LS260	.55
74LS51	.29	74LS138	1.25	74LS279	.79
74LS55	.29	74LS139	1.25	74LS367	.99
74LS73	.49	74LS151	1.25	74LS368	.99
74LS74	.49	74LS153	1.25	74LS670	3.95

BUGBOOK®

Continuing Education Series



BUGBOOKS I and II — **$17.00 per set**
by Peter R. Rony, David G. Larsen, WB4HYJ
Sold as a set these two books outline over 90 experiments designed to teach the reader all he will need to know about TTL logic chips to use them in conjunction with microprocessor systems. You'll learn about the basic concepts of digital electronics including gates, flip-flops, latches, buses, decoders, multiplexers, demultiplexers, LED displays, RAM's, ROM's, and much, much more.

BUGBOOK IIa — **$5.00**
by Peter R. Rony, David G. Larsen, WB4HYJ
This volume will introduce you to the fabulous UART chip — that all important interface between data terminals, etc., and your microcomputer. It also covers current loops, and the RS 232C interface standard. Particularly recommended for any RTTY enthusiast.

THE 555 TIMER APPLICATIONS SOURCEBOOK WITH EXPERIMENTS — **$6.95**
by Howard M. Berlin
This book shows you what the 555 timer is and how to use it. Included are over 100 various design techniques, equations and graphs to create "ready-to-go" timers, generators, power supplies, measurement and control circuits, party games, circuits for the home and automobile, photography, music and Amateur Radio.

BUGBOOK III — **$15.00**
by Peter R. Rony, David G. Larsen, WB4HYJ, Jonathan A. Titus
Here is the book that puts it all together. Besides having much valuable text there are a series of experiments in which the reader completely explores the 8080 chip pin by pin and introduces you to the Mark 80 microcomputer, a unique easily interfaced system. It is recommended that you have the background on the BUGBOOKS I & II before proceeding with BUGBOOK III.

INSTRUCTOR'S MANUAL — **$3.00**
Necessary for instruction of Bugbook I and II. Answers questions regarding experiments, suggestions for further reading, philosophy of authors approach to digital electronics. A must for self-teaching individuals.

BUGBOOK V and VI — **$19.00 per set**
by David G. Larsen, Peter R. Rony, Jonathan A. Titus
Experiments in digital electronics, 8080A microcomputer programming and 8080A microcomputer interfacing. An integrated approach to self-instructed basic digital electronics, breadboarding and 8080A interfacing/programming Bugbook VI integrates the digital concepts of Bugbook V into a treatment of 8080A microcomputer programming and interfacing. Detail & laboratory experiments included with each book.

OP AMP MANUAL — **$9.00**
An experiment guide to application of operational amplifiers. Over 25 experiments on all phases of Op Amps.

DBUG — **$5.00**
8080 interpretive debugger. A program for entering, debugging and storing assembly language programs.

CMOS-M — DESIGNERS PRIMER AND HANDBOOK — **$6.00**
Starts at basic structure of CMOS devices through integration into MSI.

COMPLETE MANUAL FOR DIGITAL CLOCKS by John Weiss and John Brooks
Familiarizes technician or hobbyist with basic theories behind digital clocks. Includes trouble shooting guides, basic characteristics of clocks, soldering techniques, clock component data sheets and construction tips. **$3.95**

DISCRETE LEDS

.125" dia.			.190" dia.		
XC209	Red	5/$1	XC111	Red	10/$1
XC209	Green	4/$1	XC111	Green	4/$1
XC209	Orange	4/$1	XC111	Yellow	4/$1
XC209	Yellow	4/$1	XC111	Orange	4/$1

.200" dia			.185" dia.			.200" dia.			.085" dia.
XC22	Red	5/$1	XC526	Red	5/$1	XC556	Red	5/$1	MV50 - Red - 6/$1
XC22	Green	4/$1	XC526	Green	4/$1	XC556	Green	4/$1	
XC22	Yellow	4/$1	XC526	Yellow	4/$1	XC556	Yellow	4/$1	INFRA-RED LED ¼" x ¼" x 1/16" Flat **5/$1.00**
XC22	Orange	4/$1	XC526	Clear	4/$1	XC556	Orange	4/$1	
SSL-22	RT	4/$1				XC556	Clear	7/$1	

DISPLAY LEDS

TYPE	POLARITY	HT	PRICE	TYPE	POLARITY	HT	PRICE
MAN 1	Common Anode-red	.270	2.95	MAN 6630	Common Anode-orange	.560	1.25
MAN 2	5 x 7 Dot Matrix-red	.300	4.95	MAN 6640	Common Cathode-orange-D.D.	.560	1.25
MAN 3	Common Cathode-red	.125	.39	MAN 6650	Common Cathode-orange ±1	.560	1.25
MAN 4	Common Cathode-red	.187	1.95	MAN 6660	Common Anode-orange	.560	1.25
MAN 7	Common Anode-red	.270	1.25	MAN 6680	Common Cathode-orange	.560	1.25
MAN 7G	Common Anode-green	.270	1.95	MAN 6710	Common Anode-red-D.D.	.560	1.25
MAN 7Y	Common Anode-yellow	.270	1.95	MAN 6730	Common Anode-red-D.D.	.560	1.25
MAN 52	Common Anode-green	.300	1.00	MAN 6740	Common Cathode-red-D.D.	.560	1.25
MAN 71	Common Anode-red	.300	1.25	MAN 6750	Common Cathode-red ±1	.560	1.25
MAN 72	Common Anode-red	.300	1.25	MAN 6760	Common Anode-red	.560	1.25
MAN 74	Common Cathode-red	.300	1.50	MAN 6780	Common Cathode-red	.560	1.25
MAN 81	Common Anode-yellow	.300	1.00	DL701	Common Anode-red ±1	.300	1.00
MAN 82	Common Anode-yellow	.300	1.00	DL702	Common Cathode-red	.300	1.25
MAN 84	Common Cathode-yellow	.300	1.00	DL704	Common Cathode-red	.300	1.50
MAN 3620	Common Anode-orange	.300	1.00	DL707	Common Anode-red	.300	1.50
MAN 3630	Common Anode-orange ±1	.300	1.35	DL741	Common Anode-red	.600	1.95
MAN 3640	Common Cathode-orange	.300	1.75	DL746	Common Anode-red ±1	.630	1.95
MAN 4610	Common Anode-orange	.300	1.00	DL747	Common Anode-red	.600	2.25
MAN 4640	Common Cathode-orange	.400	1.00	DL749	Common Cathode-red ±1	.630	1.95
MAN 4710	Common Anode-red ±1	.400	1.00	DL750	Common Cathode-red	.600	2.49
MAN 4730	Common Anode-red	.400	1.00	DL33B	Common Cathode-red	.110	.69
MAN 4740	Common Cathode-red	.400	1.00	FND70	Common Cathode (FND359)	.250	.75
MAN 4810	Common Anode-yellow	.400	1.00	FND503	Common Cathode (FND500)	.500	1.29
MAN 6610	Common Anode-orange-D.D.	.560	1.25	FND507	Common Anode (FND510)	.500	1.29

RCA LINEAR

CA3013	2.15	CA3082	2.00
CA3023	2.56	CA3083	1.60
CA3035	2.48	CA3086	.85
CA3039	1.35	CA3089	3.75
CA3046	1.30	CA3091	3.50
CA3053	1.50	CA3102	2.95
CA3059	3.25	CA3123	2.15
CA3060	3.25	CA3130	1.39
CA3080	.85	CA3140	1.25
CA3081	2.00	CA3401	.49
		CA3600	1.75

EXAR

XR-2206KB Kit $19.95 — **XR-2206KA Kit $14.95**

WAVEFORM GENERATORS

XR-205	$8.40
XR-2206CP	5.50
XR-2207CP	3.85

STEREO DECODERS

XR-1310CP	$3.20
XR-1310EP	3.20
XR-1800P	3.20
XR-2567	2.99

MISCELLANEOUS

XR-2211CP	$6.70
XR-4136	2.00
XR-1468	3.85
XR-1488	5.80
XR-1489	4.80
XR-2208	5.20

TIMERS

XR-555CP	$ .49
XR-320P	1.55
XR-556CP	1.85
XR-2556CP	3.20
XR-2240CP	4.80

PHASE LOCKED LOOPS

XR-210	5.20
XR-215	6.60
XR-567CP	1.95
XR-567CT	1.70

IC SOLDERTAIL — LOW PROFILE (TIN) SOCKETS

	1-24	25-49	50-100		1-24	25-49	50-100
8 pin	$.17	.16	.15	24 pin	$.38	.37	.36
14 pin	.20	.19	.18	28 pin	.45	.44	.43
16 pin	.22	.21	.20	36 pin	.60	.59	.58
18 pin	.29	.28	.27	40 pin	.63	.62	.61
22 pin	.37	.36	.35				

SOLDERTAIL STANDARD (TIN)

14 pin	$.27	.25	.24	28 pin	$ .99	.90	.81
16 pin	.30	.27	.25	36 pin	1.39	1.26	1.15
18 pin	.35	.32	.30	40 pin	1.59	1.45	1.30
24 pin	.49	.45	.42				

SOLDERTAIL STANDARD (GOLD)

8 pin	$.30	.27	.24	24 pin	$ .70	.63	.57
14 pin	.35	.32	.29	28 pin	1.10	1.00	.90
16 pin	.38	.35	.32	36 pin	1.75	1.40	1.26
18 pin	.52	.47	.43	40 pin	1.75	1.59	1.45

WIRE WRAP SOCKETS (GOLD) LEVEL #3

8 pin	$.40	.38	.35	22 pin	.95	.85	.75
10 pin	.45	.41	.37	24 pin	$1.05	.95	.85
14 pin	.39	.38	.37	28 pin	1.40	1.25	1.10
16 pin	.43	.42	.41	36 pin	1.59	1.45	1.30
18 pin	.75	.68	.62	40 pin	1.75	1.55	1.40

50 PCS. RESISTOR ASSORTMENTS $1.75 PER ASST.

ASST. 1	5 ea.	10 OHM / 27 OHM	12 OHM / 33 OHM	15 OHM / 39 OHM	18 OHM / 47 OHM	22 OHM / 56 OHM	1/4 WATT 5% = 50 PCS.
ASST. 2	5 ea.	68 OHM / 180 OHM	82 OHM / 220 OHM	100 OHM / 270 OHM	120 OHM / 330 OHM	150 OHM / 390 OHM	1/4 WATT 5% = 50 PCS.
ASST. 3	5 ea.	470 OHM / 1.2K	560 OHM / 1.5K	680 OHM / 1.8K	820 OHM / 2.2K	1K / 2.7K	1/4 WATT 5% = 50 PCS.
ASST. 4	5 ea.	3.3K / 8.2K	3.9K / 10K	4.7K / 12K	5.6K / 15K	6.8K / 18K	1/4 WATT 5% = 50 PCS.
ASST. 5	5 ea.	22K / 56K	27K / 68K	33K / 82K	39K / 100K	47K / 120K	1/4 WATT 5% = 50 PCS.
ASST. 6	5 ea.	150K / 390K	180K / 470K	220K / 560K	270K / 680K	330K / 820K	1/4 WATT 5% = 50 PCS.
ASST. 7	5 ea.	1M / 2.7M	1.2M / 3.3M	1.5M / 3.9M	1.8M / 4.7M	2.2M / 5.6M	1/4 WATT 5% = 50 PCS.

ASST. 8R Includes Resistor Assortments 1-7 (350 PCS.) **$9.95 ea.**

$5.00 Minimum Order — U.S. Funds Only
California Residents — Add 6% Sales Tax
Spec Sheets - 25¢ — Send 35¢ Stamp for 1978 Catalog
Dealer Information Available

NOW OPEN SATURDAYS

James ELECTRONICS

1978 CATALOG NOW AVAILABLE

1021-A HOWARD AVE., SAN CARLOS, CA. 94070
PHONE ORDERS WELCOME — (415) 592-8097
All Advertised Prices Good Thru December

HOBBY-WRAP TOOL-BW-630



$34.95
(batteries not included)

- Battery Operated (Size C)
- Weighs ONLY 11 Ounces
- Wraps 30 AWG Wire onto Standard DIP Sockets (.025 inch)
- Complete with built-in bit and sleeve

WIRE-WRAP KIT — WK-2-W

WRAP • STRIP • UNWRAP



- Tool for 30 AWG Wire
- Roll of 50 Ft. White or Blue 30 AWG Wire
- 50 pcs. each 1", 2", 3" & 4" lengths — pre-stripped wire.

$11.95

WIRE WRAP TOOL WSU-30
WRAP • STRIP • UNWRAP - $5.95

WIRE WRAP WIRE — 30 AWG

25 ft. min. $1.25 50 ft. $1.95 100 ft. $2.95 1000 ft. $15.00
SPECIFY COLOR — White - Yellow - Red - Green - Blue - Black

WIRE DISPENSER — WD-30

- 50 ft. roll 30 AWG KYNAR wire wrap wire **$3.45 ea.**
- Cuts wire to desired length
- Strips 1" of insulation **Specify** — Blue-Yellow-White-Red

REPLACEMENT DISPENSER SPOOLS FOR WD 30
Specify blue, yellow, white or red **$1.98/spool**

DIP/IC INSERTION TOOL WITH PIN STRAIGHTEN
Inserts both 14 and 16 pin packages. Pin Straightener built into Handle.
Model INS-1416 **$3.49/ea.**

HP 5082-7400 SERIES — MULTI-DIGIT

			10 or more
• ⅛" ht. • Common Cathode Red	2 Digit	$ .79	.69
• 3-5 volts @ 5 mils/second	3 Digit	.89	.79
• 7 segment Monolithic	4 Digit	.99	.89
• Dip Package	5 Digit	1.19	.99

TV GAME CHIP SET — $9.95

Includes AY-3-8500-1 Chip and 2.010 mhz crystal
(2.010 crystal — $1.95 ea/AY-3-8500-1 Chip — $8.95 ea.)

ZENERS — DIODES — RECTIFIERS

TYPE	VOLTS	W	PRICE	TYPE	VOLTS	W	PRICE
1N746	3.3	400mm	4/1.00	1N4005	600 PIV	1 AMP	10/1.00
1N751A	5.1	400m	4/1.00	1N4006	800 PIV	1 AMP	10/1.00
1N752	5.6	400m	4/1.00	1N4007	1000 PIV	1 AMP	10/1.00
1N753	6.2	400m	4/1.00	1N3600	50	200m	6/1.00
1N754	6.8	400m	4/1.00	1N4148	75	10m	15/1.00
1N959	8.2	400m	8/1.00	1N4154	35	10m	12/1.00
1N965B	15	400m	4/1.00	1N4305	75	25m	20/1.00
1N5232	5.6	500m	28	1N4734	5.6	1w	28
1N5234	6.2	500m	28	1N4735	6.2	1w	28
1N5235	6.8	500m	28	1N4736	6.8	1w	28
1N5236	7.5	500m	28	1N4738	8.2	1w	28
1N456	25	40m	6/1.00	1N4742	12	1w	28
1N458	150	7m	6/1.00	1N4744	15	1w	28
1N485A	180	10m	6/1.00	1N1183	50 PIV	35 AMP	1.60
1N4001	50 PIV	1 AMP	12/1.00	1N1184	100 PIV	35 AMP	1.70
1N4002	100 PIV	1 AMP	12/1.00	1N1185	150 PIV	35 AMP	1.50
1N4003	200 PIV	1 AMP	12/1.00	1N1186	200 PIV	35 AMP	1.80
1N4004	400 PIV	1 AMP	12/1.00	1N1188	400 PIV	35 AMP	3.00

SCR AND FW BRIDGE RECTIFIERS

C36D	15A @ 400V	SCR	$1.95
C38M	35A @ 200V	SCR	1.95
2N2328	1.6A @ 200V	SCR	.50
MDA 980-1	12A @ 50V	FW BRIDGE REC.	1.95
MDA 980-3	12A @ 200V	FW BRIDGE REC.	1.95

TRANSISTORS

MPS A05	5/$1.00			PN4250	4/$1.00
MPS A06	5/$1.00			2N4400	4/$1.00
2N918	4/$1.00			2N4401	4/$1.00
2N2219A	3/$1.00	2N3638	5/$1.00	2N4402	4/$1.00
2N2221	4/$1.00	2N3702	5/$1.00	2N4403	4/$1.00
2N2222A	5/$1.00	2N3704	5/$1.00	2N4409	5/$1.00
2N2369	5/$1.00	2N3705	5/$1.00	2N5086	4/$1.00
2N2369A	4/$1.00	2N3706	5/$1.00	2N5087	4/$1.00
2N2484	4/$1.00	2N3707	5/$1.00	2N5088	4/$1.00
2N2906A	4/$1.00	2N3711	5/$1.00	2N5089	4/$1.00
2N2907A	5/$1.00	2N3724	$ .65	2N5129	5/$1.00
2N2925	5/$1.00	2N3725	$1.00	2N5138	5/$1.00
2N3035	2/$1.00	2N3772	$2.25	2N5139	5/$1.00
2N3055	$ .89	2N3903	5/$1.00	2N5209	5/$1.00
MJE2955	$1.25	2N3904	4/$1.00	2N5210	5/$1.00
MJE3055	$1.00	2N3905	4/$1.00	2N5432	$2.00
2N3392	5/$1.00	2N3906	4/$1.00	2N5449	3/$1.00
2N3398	5/$1.00	2N4013	3/$1.00	2N5951	5/$1.00
PN3567	3/$1.00	2N4014	3/$1.00	C106B1SCR	2/$1.00
PN3568	4/$1.00	2N4123	6/$1.00	40409	$1.75
PN3569	4/$1.00	PN4249	4/$1.00	40410	$1.75

CAPACITOR CORNER

50 VOLT CERAMIC DISC CAPACITORS

	1-9	10-49	50-100		1-9	10-49	50-100
10 pf	.05	.04	.03	.001µF	.05	.04	.035
22 pf	.05	.04	.03	.0047µF	.05	.04	.035
47 pf	.05	.04	.03	.01µF	.05	.04	.035
100 pf	.05	.04	.03	.022µF	.06	.05	.04
220 pf	.05	.04	.03	.047µF	.06	.05	.04
470 pf	.05	.04	.035	.1µF	.12	.09	.075

100 VOLT MYLAR FILM CAPACITORS

.001mf	.12	.10	.07	.022mf	.13	.11	.08
.0022	.12	.10	.07	.047mf	.21	.17	.13
.0047mf	.12	.10	.07	.1mf	.27	.23	.17
.01mf	.12	.10	.07	.22mf	.33	.27	.22

+20% DIPPED TANTALUMS (SOLID) CAPACITORS

.1/35V	.28	.23	.17	1.5/35V	.30	.26	.21
.15/35V	.28	.23	.17	2.2/25V	.31	.27	.22
.22/35V	.28	.23	.17	3.3/25V	.31	.27	.22
.33/35V	.28	.23	.17	4.7/25V	.32	.28	.23
.47/35V	.28	.23	.17	6.8/25V	.36	.31	.25
.68/35V	.28	.23	.17	10/25V	.40	.35	.29
1.0/35V	.28	.23	.17	15/25V	.63	.50	.40

MINIATURE ALUMINUM ELECTROLYTIC CAPACITORS

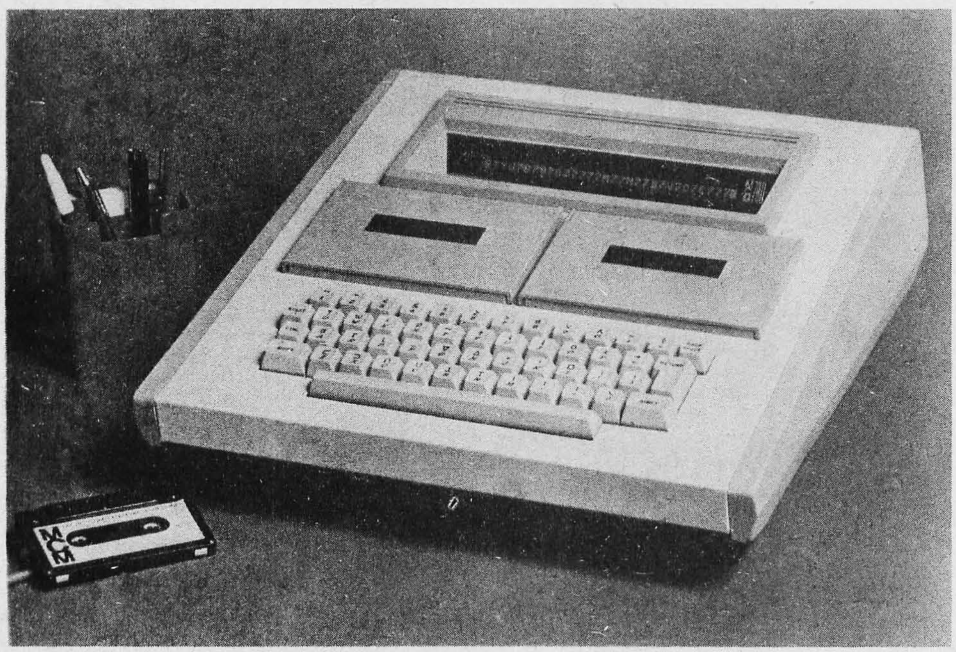
Axial Lead				Radial Lead			
.47/50V	.15	.13	.10	.47/25V	.15	.13	.10
1.0/50V	.16	.14	.11	.47/50V	.16	.14	.11
3.3/50V	.15	.13	.10	1.0/16V	.15	.13	.10
4.7/25V	.16	.14	.12	1.0/25V	.16	.14	.11
10/25V	.15	.13	.10	1.0/50V	.16	.14	.11
10/50V	.16	.14	.12	4.7/16V	.15	.13	.10
22/25V	.17	.15	.12	4.7/25V	.15	.13	.10
22/50V	.24	.20	.18	4.7/50V	.16	.14	.11
47/25V	.19	.17	.15	10/16V	.14	.12	.09
47/50V	.25	.21	.19	10/25V	.15	.13	.10
100/25V	.24	.20	.18	10/50V	.16	.14	.12
100/50V	.35	.30	.28	47/50V	.24	.21	.19
220/25V	.32	.28	.25	100/16V	.19	.15	.14
220/50V	.45	.41	.38	100/25V	.24	.20	.18
470/25V	.33	.29	.27	100/50V	.35	.30	.28
1000/16V	.55	.50	.45	220/16V	.23	.17	.16
2200/16V	.70	.62	.55	470/25V	.31	.28	.26

Circle 67 on inquiry card.

What's New?

SYSTEMS

MCM's Desk Top APL Computer



MCM Computers Inc has announced the reduction in price of its System 700, Model 782 APL computer to $4,950, an offer explicitly aimed at the personal computing user.

The stand alone unit is smaller than an office typewriter, weighs only 20 pounds (9 kg), and features a full APL language interpreter interchangeable with the latest implementations running on MCM's larger System 800 and comparable to APL.SV implementations running on large scale systems such as the IBM 370. The unit also contains file handling and virtual memory (AVS) operating systems, and provides up to 100 K bytes of virtual memory swapping on one of its two built-in cassettes. A total of 200 K bytes of randomly accessible magnetic tape storage is available between the two cassette drives. This "external" memory is in addition to the built-in 8 K bytes of programmable memory and 32 K bytes of read only memory. The MCM 782 also incorporates a battery backup system that automatically saves the user's workspace in case of power loss. Other technical features incorporated into the machine include dynamic memory allocation of one to eight bytes for integer data to conserve memory, total system overhead of less than 500 bytes, and extended prompting facility for interactive English language conversational programs. The unit has an integrated plasma alphanumeric display, full 46 key input and a bus structure to allow interface to the company's other peripheral products. Attachments supported by the MCM unit include floppy disk, printer, plotter, card reader and a serial communications interface that provides RS232C connection to a host of standard devices in the general computing market.

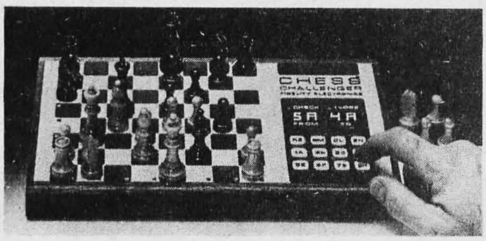
The RS232C interface, called the SCI-1200, is also being announced for $650. The unit provides programmable protocol and line speed for serial connection to any 4 to 8 bit serial device. It is fully supported by software in read only memory and provides an ability to connect any ASCII, EBCDIC, correspondence or other devices. Printers such as Teletypes and DECwriters can be connected via this SCI-1200, as can an acoustic coupler for communications with other systems, at speeds of up to 1200 bps.

MCM provides a number of software packages for use on its system, including math, statistics, finance, plotting, general ledger, order entry, billing, production planning, dynamic purchasing, management problem solving, and so on. A text editor package called Text/700 is also available. A computer games package is available for $100, which includes blackjack, hangman, ecology, and math drills for younger users.

The MCM 782 is being marketed direct from MCM to end users at a price of $4,950, FOB Fort Lee NJ for shipments in the US, and FOB Kingston Ontario for shipments in Canada. All units are factory warranted for a period of 90 days. Deliveries are presently available from stock. All prepaid orders or inquiries should be directed to MCM's headquarters in either the US or Canada. Contact MCM Computers Inc, 2125 Center Av, Fort Lee NJ 07024, (201) 944-2737, or, in Canada, MCM Computers Ltd, 6700 Finch Av W, Suite 600, Rexdale, Ontario CANADA M9W 5P5, (416) 675-1353.■

Circle 545 on inquiry card.

A Chess "Mate" from Chafitz



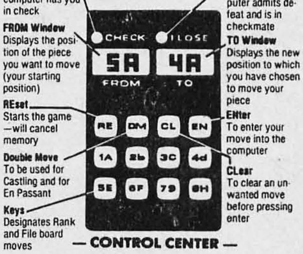
Chafitz has introduced a very interesting device: a $200 Chess Challenger designed to play chess against human competition at a fairly sophisticated level.

The unit is microprocessor based. The physical package consists of a rectangular enclosure containing the electronics, on top of which is a chess board, a calculator-like keyboard and an LED display. The user enters moves using a simple coordinate code system, and the computer displays its moves on the LED display. Any regulation move can be made, including castling (king or queen side) and capturing en passant.

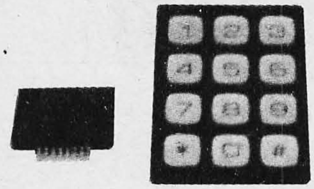
If the user improves to the point where he or she can consistently beat the Chess Challenger, the unit can be returned to have a more sophisticated set of algorithms entered by the manufacturer.

The device is intended primarily for the chess novice, of course, but unless you're Bobby Fisher (who claims to have beaten the unit every time in the #1 issue of *Computer Chess Newsletter*) you should find it to be of interest. Hackers who are also chess freaks will be inspired to try their skills at designing and programming chess playing algorithms of their own.

For more information, contact Chafitz, 1055 First St, Rockville MD 20850, (301) 340-0200.■

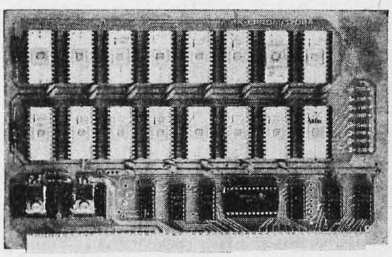


TOUCH TONE ENCODER KIT



Simplicity itself to complete. No other parts required, no crystal required. The back of the touch pad has etched & drilled PC board and you solder the encoder chip to it. Add your own small speaker & 9 volt battery and you are done. A touch of the pad produces the proper tone signal from the speaker. We furnish schematic and instructions.

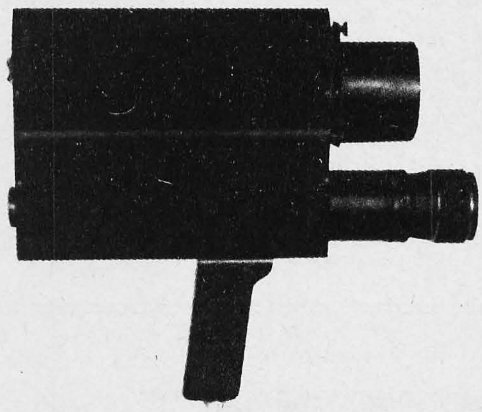
SP-149-B $12.95



VIATRON CASSETTE DECKS

The computer cassette deck alone $35. Set of 2 boards read/write amp & serve control boards of this deck. $40.00

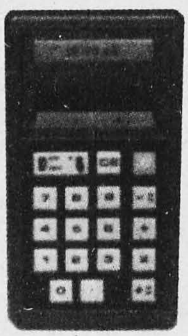
IR NIGHT VIEWER $199.00



Custom made, complete with light source & viewer in one piece. Comes with carrying strap. Ready to operate with 6 volt lantern battery. Guaranteed by the manufacturer. See in total darkness. Great for scientists, viewing nocturnal animals & birds, criminal investigation. . .observe without being observed, and a ball for just plain snooping!!!! Sorry to say but no shipments to Calif. (lens may vary slightly from pic)

SPL-21 $199.00

FACTORY REJECT CALCULATORS



All more or less as shown. Some have "K" some don't. Not a cheapy. Probably sold for around $25.00. We check a dozen or so and we found they seem to work for the most part. Runs on snap-on 9V battery. Some 4, some 5 function. We include a reprint of How to fix calculators from one of the magazines. Sold AS IS.

Ship wgt. 1 lb SP-253A $3.00

WIRE WRAP WIRE

TEFZEL blue #30 Reg. price $13.28/100 ft. Our price 100 ft $2.00; 500 ft $7.50.

MULTI COLORED SPECTRA WIRE

Footage		10'	50'	100'
8 Cond.	#24	$2.50	9.00	15.00
12 "	22	3.00	11.00	18.00
14 "	22	3.50	13.00	21.00
24 "	#24	5.00	20.00	30.00
29 "	22	7.50	28.00	45.00

Great savings as these are about 1/4 book prices. All fresh & new.

TOUCHTONE ENCODER CHIP

Compatible with Bell system, no crystal required. Ideal for repeaters & w/specs. $6.00

CHARACTER GENERATOR CHIP

Memory is 512x5 produces 64 five by seven ASCII characters. New material w/data $6.00

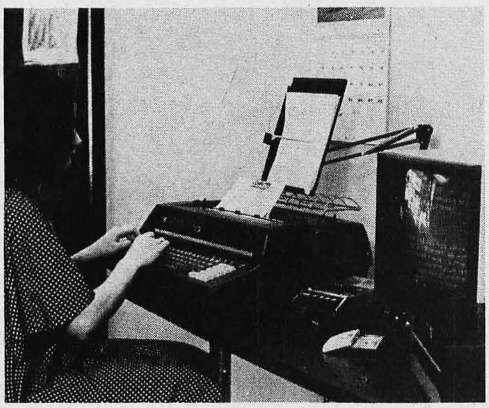
Meshna

Please add shipping cost on above. Minimum order $10

FREE CATALOG SP-10 NOW READY

P.O. Box 62, E. Lynn, Massachusetts 01904

A Real Word Processor



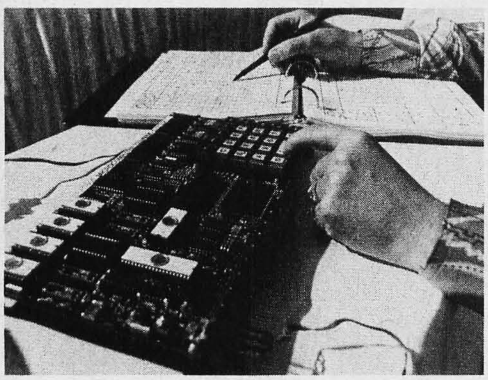
Computer Power and Light Inc, 12321 Ventura Blvd, Studio City CA 91604, (213) 760-0405, has announced this "commercial quality" microcomputer based word processing system for under $6000, complete. Based on the firm's COMPAL-80 computer and Xerox Corporation's Diablo 1620 daisy wheel printer, it is said to contain features previously found on systems costing $20,000 or more.

Features of the machine include text editing using a large CRT display; insertion or deletion of text, and the ability to move blocks of text anywhere; variable speed scrolling of entire text on the CRT, forward or backward; ability to search for all occurrences of a specific word or group of words, and replacement with alternative word or words; storage and retrieval of finished text on low cost Philips audio cassettes at the rate of 240 characters per second; a variety of printing options including variable line length; 1 to 5 spaces between lines; variable character spacing; presettable page headings; page numbering; and right and left margin justification using the Diablo's unique character spacing routines: no extra blanks are inserted in your text, nor is there any need for hyphenation.

This interesting application system is available from Computer Power and Light, 12321 Ventura Blvd, Studio City CA 91604 or 7878 Clairemont Mesa Blvd, San Diego CA 92111. 4 year lease plans and bank financing are available.■

Circle 539 on inquiry card.

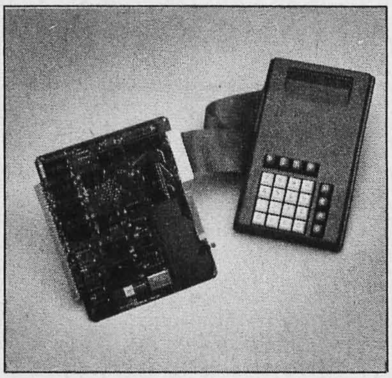
ICS Microcomputer Trainer



A complete self-study microcomputer training course including an 8080 based single board computer has been introduced by Integrated Computer Systems Inc, 4445 Overland Av, Culver City CA 90230, (213) 559-9265. The computer includes 512 bytes of CMOS memory, a 1 K byte monitor in electrically erasable PROM, a keyboard and LED display, and a prototyping area for audio cassette, RS-232, current loop or other interface circuits. The board is accompanied by a 650 page *Microcomputer Training Workbook* which teaches the 8080 instruction set, basic programming and hardware design techniques. The entire package, minus a power supply, is offered for $545.■

Circle 540 on inquiry card.

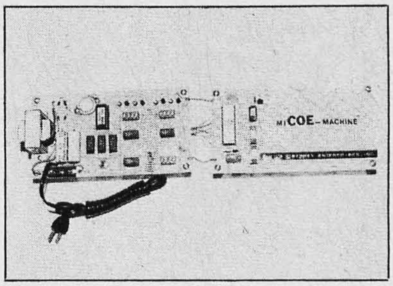
Pre-assembled Motorola MEK6800D2



A fully assembled version of the Motorola MEK6800D2 kit, requiring only a 5 V, 1 A power supply, is available from Audio Engineering, 121 Wisconsin NE, Albuquerque NM 87108, (505) 255-6451. The SY1-068 features a stand for the processor board and an attractive case for the keyboard and LED display. The processor board includes 256 bytes of memory, a 1 K byte ROM monitor, cassette interface, and parallel IO. The SY1-068 is priced at $269. Accessories include the keyboard and display case at $12.50, an extra 128 bytes of memory at $7.50, and a power supply kit with a 60 Hz clock at $29.95.■

Circle 543 on inquiry card.

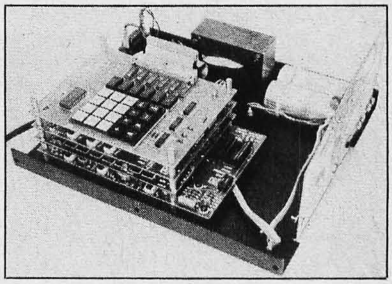
COSMAC Based Microcomputer Kit



This kit, based on the RCA COSMAC 1802 processor, is designed to be assembled by the user by wire wrapping on two pieces of Micro Vectorboard. The kit includes a 5 V, 1 A power supply, eight toggle switches and LEDs, processor integrated circuit and clock, and 256 bytes of memory. Accessories such as a cabinet and memory are planned. The kit is priced at $90. Wire wrap sockets, wire and tool are an additional $10, available from Child Odyssey Enterprises Inc, POB 137, Alamogordo NM 88310, (505) 434-1065.■

Circle 541 on inquiry card.

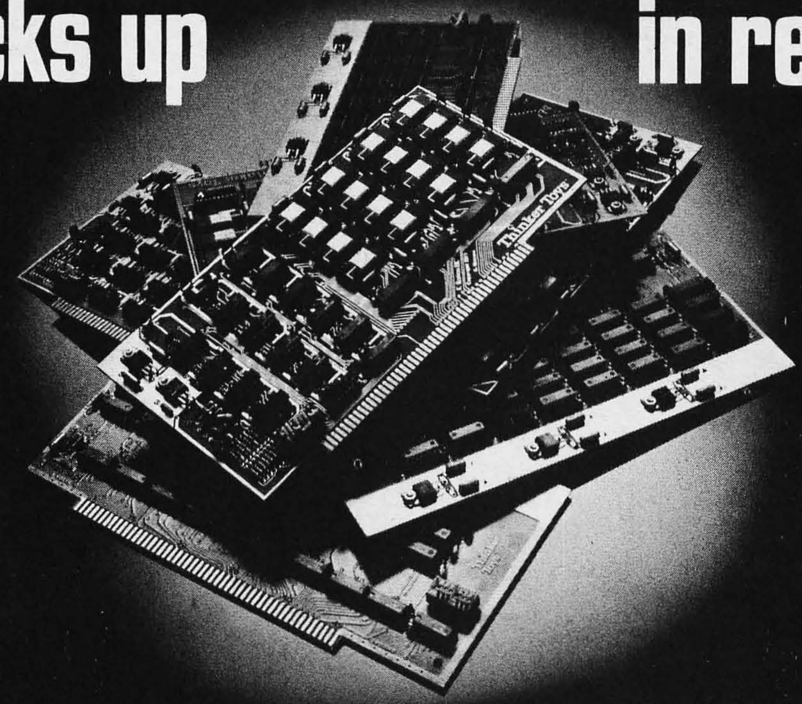
Z-80 Based System Features Support Boards



The new Z-80 based Mike 8 system features compatibility with most of the boards already available for the 8080 based Mike 3. The full blown Model 882 version of the Mike 8 includes a CPU board, 4 K bytes of memory, a 1 K byte monitor program in ROM, and a "console board" with a keyboard and LED display. The package also includes an EROM programmer, a blank 2708 EROM, and an ultraviolet lamp for EROM erasure. The Model 882 is mounted on a base with a switching regulated power supply. An extensive manual and a book entitled *Microcomputer Design* complete the package, which sells for $895. Smaller Z-80 based systems are also offered, starting at $495. The book *Microcomputer Design* is available separately for $14.95. The Mike 8 is available through Semiconductor Specialists, the industrial distributor based in Elmhurst IL, as well as the "MPU Shops" which Semi Specs is opening at many of its branch offices, or from the manufacturer, Martin Research, 3336 Commerical Av, Northbrook IL 60062, (312) 498-5060.■

Circle 542 on inquiry card.

Now low-cost memory stacks up in reliability!



Introducing a new generation of ECONORAM* dynamics with SynchroFresh™ reliability

Meet ECONORAM* III with SynchroFresh™, the 8Kx8 dynamic memory for S-100 bus computers that really works. And uses less than half the power of static designs. And costs just $149 for an assembled 8K.

Unlike previous attempts at building a low-cost dynamic memory, ECONORAM* III is entirely reliable ... because of SynchroFresh™, a new approach to memory refresh that is simple, elegant and totally effective.

SynchroFresh™ was invented by George Morrow, designer of the original ECONORAM*. Instead of arbitrarily interrupting your CPU to perform memory refresh cycles, Morrow designed SynchroFresh™ to weave refresh invisibly into the natural timing of the S-100 bus. SynchroFresh™ circuitry simply monitors your computer's machine states, utilizing all of the normal opportunities for memory refresh. It's that simple.

And simplicity means reliability and dramatically lower cost. That's why a SynchroFresh™ design was chosen for the first ECONORAM* dynamic, to follow in the footsteps of the largest-selling static memories for personal computers.

ECONORAM* III with SynchroFresh™ is an 8Kx8 dynamic board, configured as two individually addressable 4K blocks for flexibility. It is available assembled, tested and warranteed for one full year for just $149. This unprecedented warrantee offers a full refund of purchase price if ECONORAM* III does not run reliably with your S-100 CPU—evidence of our confidence in its performance.

It is also available as a kit with complete assembly instructions and documentation for $159.

ECONORAM* III with SynchroFresh™, in assembled or kit form, may be ordered directly from Thinker-Toys™. Write 1201 10th Street, Berkeley CA 94710 or call (415) 527-7548. Call BAC/MC orders toll-free to 800-648-5311. Or ask your computer store to order it for you.

NEW LOW PRICE

$149

8K assembled, tested, warranteed 1 year

A product of Morrow's Micro-Stuff for

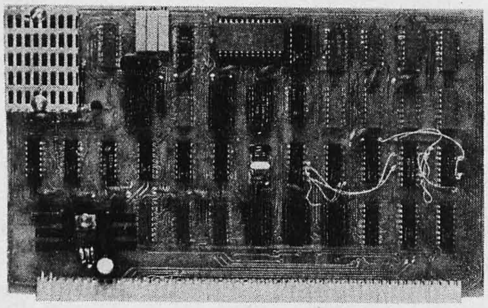
Thinker Toys™

*ECONORAM is a trademark of Godbout Electronics.

Circle 78 on inquiry card.

PERIPHERALS

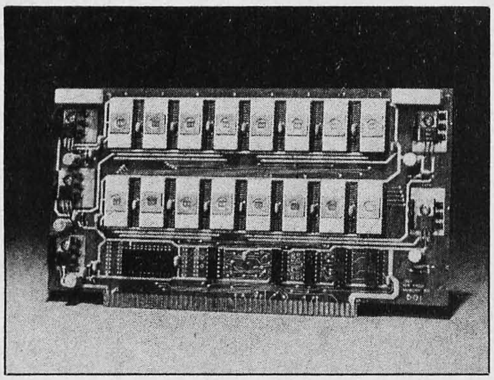
Video Board for 6800 Systems



A printed circuit board based on Alfred Anderson's article "Build This Video Display Terminal," November 1976 BYTE, page 106, is available from F & D Associates, Box 183, New Plymouth OH 45654, (614) 385-2023. The circuit generates a 16 by 32 character display and can switch between two 512 byte pages of memory. The board is designed to plug into the SwTPC (SS-50) bus, but could be interfaced to other 6800 systems. Space is provided on the board for mounting an ATV Research Pixie-Verter (RF modulator). The board and construction hints are offered for $29 plus $2.50 shipping.■

Circle 546 on inquiry card.

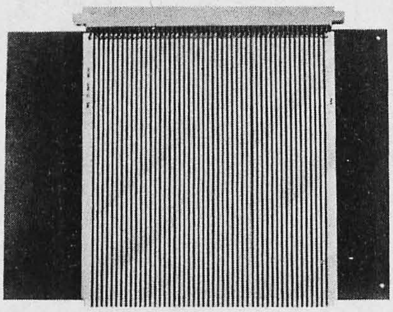
16 K of EROM on One Board

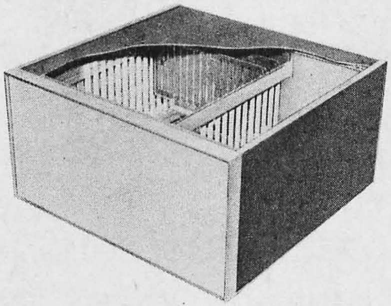


This memory board is designed to hold up to 16 of the popular 2708 1 K byte EROMs, which have lately dropped to a very affordable price. Unused 4 K sections of the board can be disabled to avoid consuming excess memory address space. A wait state feature is provided for fast Z-80 based systems. The complete kit for an Altair (S-100) bus computer is available for $85, minus the 2708 EROMs, from IBEX, 1010 Morse Av, Suite 5, Sunnyvale CA 94086, (408) 739-3770.■

Circle 550 on inquiry card.

New Enclosures and Card Extender

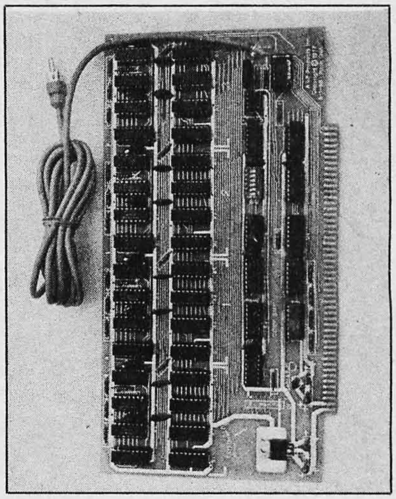




Enclosures for Altair (S-100) bus compatible systems are now available from Vector Electronic Co Inc, 12460 Gladstone Av, Sylmar CA 91342, (213) 365-9661. The enclosure dimensions, 17.9 by 9.0 by 17.1 inches (45.5 by 22.9 by 43.4 cm), allow room for up to 21 cards, and plastic card guides are provided for 12 cards. Adjustable slots are provided for a mother board. Slide-in top, bottom and side panels free of screws and fasteners make for an attractive finish. Optional accessories include a prepunched rear panel with ten holes for DB25 connectors. The VP1 ($128.30) uses the Altair configuration of side to side card orientation with power supply in the rear, while the VP2 ($134.30) has the IMSAI configuration of front to back card orientation with power supply on the right side. Also available from Vector is the 3690-12 7.5 inch (19.0 cm) card extender, assembled with edge connector for $25.■

Circle 548 on inquiry card.

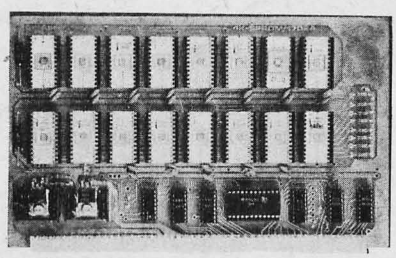
First of a Series of Music Boards



The 10-5-9 and 10-5-10 quad chromatic pitch generator boards are designed to be a low cost start in computer controlled music generation. The single board pitch generator produces one to four tones simultaneously; two boards can be used to produce eight simultaneous tones in stereo. Each of the four tones are separately controlled and can produce any of 96 tones which form an 8 octave range. This covers the entire standard piano range, plus eight higher pitches. Special connections allow later expansion with accessory boards to control various sound parameters. Using the optional on board crystal oscillator or a 2 MHz source (external or pin 49 on the S-100 bus) all pitches are within 0.1% of the A-440 Hz standard. The 10-5-9 is S-100 compatible, and the 10-5-10 is compatible with parallel output ports. Kit prices for both versions range from $111 to $159 (depending on the number of simultaneous tones), and the assembled price is $185. The oscillator is an additional $16. Available for product evaluation: data sheet (free), demonstration record ($1), and owner's manual ($3 plus $1 postage). Contact ALF Products Inc, 128 S Taft, Denver CO 80228, (303) 234-0871.■

Circle 547 on inquiry card.

4 K 1702A EROM Board



A 4 K byte 1702A-based EROM memory board for the SwTPC 6800 and similar systems is available from Aptec Inc, POB 15296, Tulsa OK 74115. An accessory clock stretcher board, based on Jerry Henshaw's article "Stretch That 6800 Clock," December 1976 BYTE, page 42, is offered to accommodate the relatively slow 1702As. A complete kit minus the 1702As is priced at $87.50, while the printed circuit board and connector alone are $27.50. The clock stretcher kit is $6.25, or $2.50 for the board.■

Circle 549 on inquiry card.

Classified Ads

FOR SALE: Used 83 pin cards made by Computer Entry Systems, with many good 7000, 7100 and 7400 series chips with slight cosmetic imperfections. All boards $6.50 to $13, depending on complexity, or trade for early BYTEs. Ivan Reeder, 414 Yeonas Dr, Vienna VA 22180.

FOR SALE: IMSAI 8080 with 22 slot and fan, VB-1 video board, Morrow cassette with extras, two IO Cybercom board, two 4 K programmable memory boards. All less than one year old and total assembled, value $1,632. Lack of time for hobby. Will sell for best offer over $1,200. Can ship UPS. James R Poole K4VBH, POB 268, Americus GA 31709.

FOR SALE: BYTE Vol 1 #1 to date, excellent condition, best offer. Will ship securely packed and insured. Henry Stevens, POB 417, Alton NH 03809. (603) 875-5550.

FOR SALE: Assembled, one RM 8080 CPU board with all chips $145; one SwTPC TVT1 debugged and running $100; one TVT1 not running $50; two Digital Group 8 K memory without memory chips $65 each; four 1702s EPROM $12 each; one 5204 EPROM 750NS $8; two 82S129 PROMs $2.50 each. Send SASE for information. David Parrish, Rt 1, Box 151, Lincolnton GA 30817.

FOR SALE: Friden Flexowriter, complete electric typewriter with attached 8 level tape punch and 8 level tape reader. Provision for input from external source, and output to external punch or other data equipment, excellent condition, only $195. Nicely styled operating desk for above, includes motorized tape winder and also tape feed reel, plus cables for remote input and output, $39. Jim Cooper, POB 73, Paramus NJ 07652.

FOR SALE: Zilog Z-80 MCB board. Contains Z-80 microprocessor, 4 K bytes programmable memory, sockets for 4 K bytes ROM, RS-232/TTY interface, two 8 bit IO ports, 4 channel counter/timer and more. Assembled, tested and working. Includes power supply, 1/2 K byte monitor PROM, and all manuals, $475. Brian Rosen, 1127 Kentwood, San Jose CA 95129.

WANTED: A true proportional spacing typing machine (eg: IBM composer) to use as a microcomputer input output device. Write J Williams, 2415 Ansdel Ct, Reston VA 22091.

FOR SALE: One each IMSAI cassette interface. Make offer. Write J Williams, 2415 Ansdel Ct, Reston VA 22091.

FOR SALE: IBM 1401 CPU, 1402 card reader/punch, two 7330 tape drives, and all maintenance manuals. Qualifies for IBM maintenance agreement. Best offer over $1000. You pay shipping. Will consider trade for a Z-80 based microcomputer. William P Ruf, 9514 W 104th Ter, Overland Park KS 66212. (913) 888-9213.

Readers who have equipment, software or other items to buy, sell or swap should send in a clearly typed notice to that effect. To be considered for publication, an advertisement should be clearly noncommercial, typed double spaced on plain white paper, and include complete name and address information. These notices are free of charge and will be printed one time only on a space available basis. Insertions should be limited to 100 words or less. Notices can be accepted from individuals or bona fide computer users clubs only. We can engage in no correspondence on these and your confirmation of placement is appearance in an issue of BYTE.

Please note that it may take three or four months for an ad to appear in the magazine.■

FOR SALE: Maze generation program written in FORTRAN. This program generates unlimited page mazes with only one solution each. The computer gives the solution, too. A source listing is available with implementation documentation for $25. It is 735 lines long, mostly comments on usage. If you want to see a sample run, please send $1 to cover printing and computer time charges to below address. Every maze is different! Philip Hunt, 83 Cedar Hills, Cambridge OH 43725.

FOR SALE: Tape reader device. Originally used to read three frames of paper tape at a time and convert the data to 15 level parallel output. Housed in a 5 foot rack cabinet with a 3/4 door and casters are: 1 each Digitronics 2500 300 cps 8 level reader and 4566ALCR spooler; 20 slot 36 pin card cage; 12 Navcor logic boards, 1 extender; 1 Navcor ± 12 V @ 2 A, -100 V @ ¼ A power supply, metered; 1 chassis with relays, etc, original cost $9,500; now $600 (you pick up or freight; 200 lbs) Mike Gerow, 244 W 1st Av, Roselle NJ 07203. (201) 241-3200; (201) 334-7246 nights.

NEED: Information on General Automation SPC 12/12 with 16 K core. Any software would be appreciated. Manual C Martinez, 7706 W Gregory St, Chicago IL 60656. (312) 631-6623.

FOR SALE: IBM 604 Electronic Calculator with IBM 521 Reader Punch. 604 has over 1,000 vacuum tube circuits with power supplies producing +150 VDC at 8 A, 100 VDC at 2.5 A, 250 VDC at .05 A, +75 VDC at 1.0 A, and 175 VDC at .25 A. 521 reads and punches 80 column cards. It has two sets of read brushes and punch unit. Lots of potential for 604 components. 521 would make a good card IO with proper interface. Will take $500 for both. You pay transportation. George Kennedy, 2101 Church St, Selma AL 36701. (205) 872-8263.

WANTED: Poly 88 mainframes (2) kit or assembled, will buy or trade. Contact Wallace Tufen, 1504 Blackthorn Dr, Glenview IL 60025. (312) 724-7828.

FOR SALE: HP-65 fully programmable pocket calculator complete with all 14 accessory application Pacs (Finance, Math(2), Stat(2), EE(2), Chem Engr, Stress Analysis, Surveying Machine Design, Medical, Aviation, and Navigation) and other accessories. Total price $350. Joe Kwasnieski, Apt 44, 13700 37th Av S, Seattle WA 98168. (206) 243-6967 evenings.

FOR SALE: Digital Group 8 K memory boards. Uses 450 ns low power memory chips, $225. Contact Roger Schoenmeyer, 7425 Treon Pl, Dayton OH 45424. (513) 233-2355.

FOR SALE: HP-9810 desktop programmable calculator with HP-9862 plotter. Calculator is complete with fully expanded optional memories for both data and program, 16 column alphanumeric thermal printer, Mathematics ROM, magnetic card storage, and four IO parts. Plotter is a high performance X-Y graphics plotter capable of plots up to 11 by 17 inches. Includes optional plotter control that permits automatic scaling, axis drawing, and alphanumeric plotting and labeling. Privately owned, in perfect condition; asking $3500. Joe Kwasnieski, Apt 44, 13700 37th Av S, Seattle WA 98168. (206) 243-6967 evenings.

SWAP: Scott Model 433 Digital FM Stereo Tuner, frequency synthesized using PLL, digital readout, card programmed (easily modified for computer control), several scanning modes, etc, originally cost $650. I want to trade for a microcomputer or related equipment. Prefer 6502 or 6800 based system, but will consider anything. Make me an offer I can't refuse. Schematics available for the tuner. Call days (617) 373-9171. Ray Jorgenson, RFD 3, Box 186, Plaistow NH 03865.

MICRODATA REALITY: Are there any other computer hobbyists using this system? If so, I'd like to say hello, swap notes and programs, etc. Would also like to know where to buy a 4 or 8 way video terminal interface card and other peripherals for Microdata Reality (Model 1600 processor). Jack Hardman, 140 Forest Av, Glen Ridge NJ 07028. (201) 429-8880.

FOR SALE: Digideck™ cassette tape drive (uses standard audio type cassette) complete with motor control, read write, and parallel to serial conversion electronics. Also, 100 plus page service manual and parts catalog included. All diagrams are very readable. These six drives appear perfect but may have a problem or two. Price $145 each plus postage. Jim Beistle, 3728 Wilkie Way, Fort Worth TX 76133.

FOR SALE: Source listing of program to calculate 1977 Federal Income Tax. Program will run in about 3 K. The program has 1040 line numbers to identify the calculated results when displayed on a CRT terminal. It is also formatted to fill out the form 1040 when the results are outputted to a printer. Send $14.50 for copy of source listing and user operating instructions. C R Lufkin, 315 Dominion Dr, Newport News VA 23602.

FOR SALE: MITS Altair 8800a computer with 1 K of memory. Works fine. $575, or best offer. Kurt Barbee, 2580 SW 3rd St, Corvallis OR 97330.

FOR SALE: SwTPC 6800 computer, 8 K programmable memory, AC-300 cassette interface, CT-1024 TV terminal with cover, connecting cables and software. All equipment assembled and tested, $850. Dave Toruta, 4941 Mable, Corpus Christi TX 78411. (513) 854-2747.

FOR SALE: Three IBM 2311 disk drives. Still eligible for IBM maintenance contract. $450 each plus shipping. Wanted: Information for interfacing same with 8080 system. Bob Stek, 19 Mayfield Rd, Regina, Saskatchewan CANADA. (306) 523-7184.

FOR SALE: Okidata CP-110 printer; loaded with such options as: RS-232 interface (110 to 9600 baud), upper and lower case character set, tractor feed option, electronic top-of-form option, and complete on board self-test electronics included. Cost over $1900 new, and has only been used twice; still in original factory carton. Must sacrifice for $1600 and you pay shipping. Contact Don Cheeseman, POB 5534, San Antonio TX 78201. (512) 699-6880.

FOR SALE: Complete first year of BYTE, issues 1 thru 16. All packed and ready to ship. I pay postage. First $35 gets them. R Peters, Lafayette Ln, Norfolk MA 02056.

FOR SALE: A complete set of circuit boards for the TVT-1, as presented in the September 1973 issue of *Radio-Electronics*; $20 plus postage. Also available: a 45 key keyboard (less encoder), $15 plus postage. Write Warren Spivack, 6625 Av M, Brooklyn NY 11234. (212) 763-7237.

WANTED: Programs for HP-25 not including those listed in original manual. Will exchange new ones for the ones I have obtained. S Hamilton, Rt 2, Box 1022, Bainbridge GA 31717.

SWAP: Two Scott aircraft oxygen units; each unit supplies two persons with O_2. With two new masks, hoses and connectors. Also, a skydiver O_2 unit with mask, hose and connector. Want: ADM-3 or comparable CRT and keyboard, RS232C and compatible with standard microcomputer, or, want: Transistorized oscilloscope, minimum 10 MHz bandwidth. F Lawrence, 533 Riverview Rd, Swarthmore PA 19081.

CONGRATULATIONS! This could be your lucky day, if you are looking for a line printer. I have one complete 100 cps PRINTEC 100 serial, impact, line printer (ASCII parallel) and one nearly complete backup unit, with extras for all boards. Unit includes vertical formatting unit, VFU punch, extra VFU tape and maintenance manuals. Adjustable pin feed accommodates paper widths from 3 inches to 14 7/8 inches. One unit is in working order; the other good for parts. Send your best offer to David E Fulton, POB 116, Port Ewen NY 12466, or call (914) 331-1442.

NEW COMPUTER INTERFACE BOARD KIT

$49.00

Our new computer kit allows you to interface serial TTL to RS 232 and RS 232 to TTL. There are four of these supplied with the kit, so you can run up to four devices on one TTL or four separate TTL to RS 232 devices.

Typical use: You can use your computer ports to run an RS 232 printer, video terminal and two other RS 232 devices at once, without constantly connecting and disconnecting your terminals.

Example: Out store to printer — Voltage requirement +5V and ±5V or ±12 V depending on your RS 232 device.

We supply — board, connectors, documentation and components. Sorry, we do not supply case or power supply.

WHERE IT MAKES SENSE, MAY BE USED WITH ANY 8080, 6800, Z80 or F8 COMPUTER

GENERAL PURPOSE COMPUTER POWER SUPPLY KIT

This power supply kit features a high frequency torroid transformer with switching transistors in order to save space and weight. 115V 60 cycle primary. The outputs with local regulators are 5V to 10A, in one amp increments, −5V at 1A, ±12V at 1A regulators supplied 6 340T-5 supplied.

$79.00

UNIVERSAL 4K MEMORY BOARD KIT $74.50

This memory board may be used with the F8 and with minor modifications may be used with KIM-1µp.

32-2102-1 static RAM's, 16 address lines, 8 data lines in, 8 data lines out, all buffered. On-board decoding for any 4 of 64 pages, standard 44 pin, .156″ buss.

F8 EVALUATION BOARD KIT WITH EXPANSION CAPABILITIES

A fantastic bargain for only with the following features:

$99.00

- 20 ma or RS 232 interface
- 64K addressing range
- Program control timers
- 1K of on-board static memory
- Built in clock generator
- 64 Byte register
- Built-in priority interrupts
- Documentation
- Uses Fairbug PSU

FOR FAIRBUG 4K F8 BASIC ON PAPER TAPE $25.00

2708-8K EPROM $15.75
2522 STATIC SHIFT REG .... $ 1.95
2513 CHARACTER GEN .... $ 6.75
2518-HEX 32 BIT SR .... $ 3.50
2102-1 1024 BT RAM .... $ 1.19
5280 4K DYNAMIC RAM .... $ 4.75
5202A UV PROM .... $ 6.95
1101A-256 BIT RAM .... $ .75
2107B .... $ 3.75
MK4008P .... $ 1.95
1702A UV PROM .... $ 4.95
5204 4K PROM $10.95
82S23 .... $ 1.95
AY-5-1013 UART .... $ 6.95

MINIATURE MULTI TURN TRIM POTS 100, 500, 1K, 2K, 5K, 10K, 25K, 50K, 100K, 200K, 500K 1 Meg. $.75 each. .... 3/$2.00

MULTI-TURN TRIM POTS Similar to Bourns 3010 style 3/16″x5/8″x1 1/4″, 50, 100, 1K, 10K, 50K ohms! $1.50 ea. .. 3/$4.00

LIGHT ACTIVATED SCR's TO-18, 200V 1A .... $ 1.10

TRANSISTOR SPECIALS

2N3585 NPN Si TO-66 .... $ .95
2N3772 NPN Si TO-3 .... $ 1.00
2N456A PNP GE .... $ .75
2N4908 PNP Si TO-3 .... $ 1.00
2N6056 NPN Si TO-3 Darlington .... $ 1.70
2N5086 PNP Si TO-92 .... 4/$ 1.00
2N4898 PNP TO-66 .... $ .60
2N404 PNP GE TO-5 .... 5/$ 1.00
2N3919 NPN Si TO-3 RF .... $ 1.50
MPSA 13 NPN Si TO-92 .... 3/$ 1.00
2N3767 NPN Si TO-66 .... $ .70
2N2222 NPN Si TO-18 .... 5/$ 1.00
2N3055 NPN Si TO-3 .... $ .60
2N3904 NPN Si TO-92 .... 5/$ 1.00
2N3906 PNP Si TO-92 .... 5/$ 1.00
2N5296 NPN Si TO-220 .... $ .50
2N6109 PNP Si TO-220 .... $ .55
2N3638 PNP Si TO-5 .... 5/$ 1.00
2N6517 NPN TO-92 Si .... 3/$1.00

C/MOS (DIODE CLAMPED)

74C10– .22	4012– .22	4023– .22	4046–2.25
74C193–1.50	4013– .40	4024– .75	4049– .40
4001– .22	4015– .95	4025– .22	4050– .40
4002– .22	4016– .40	4027– .40	4055–1.50
4006–1.20	4017–1.05	4028– .88	4066– .80
4007– .22	4018–1.00	4029–1.10	4071– .27
4009– .42	4019– .25	4030– .22	4076–1.05
4010– .42	4020–1.05	4035–1.10	
4011– .22	4022– .95	4042– .78	

IN 4148 (IN914) .... 15/$1.00

MCA 81 OPTICAL LIMIT SWITCH. $1.50

LED READOUTS

FND 359 C.C. .4″ $ .50
FND 70 C.C. .4″ $ .55
FND 503 C.C. .5″ $ .85
FND 510 C.A. .5″ $ .85
DL704-.3″ C.C. $ .95
MAN-7-.3″ C.A. $ .95
NS 33-3 dig. array $ .75
DL 747 C.A. .6″ $1.95

PRINTED CIRCUIT BOARD

4-1/2″x6-1/2″ SINGLE SIDED EPOXY BOARD 1/16″ thick, unetched $.60 ea. .... 5/$2.60

7 WATT LD-65 LASER DIODE IR $8.95

2N 3820 P FET .... $ .45
2N 5457 N FET .... $ .45
2N2646 .... $ .45
ER 900 TRIGGER DIODES .. 4/ $1.00
2N 6028 PROG. UJT .... $ .65

CCD202-100 x 100 charge coupled device image sensor $145.00

MCM6571A 7 x 9 upper & lower case character generator $10.75

VERIPAX PC BOARD

This board is a 1/16″ single sided paper epoxy board, 4½″x6½″ DRILLED and ETCHED which will hold up to 21 single 14 pin IC's or 8, 16, or LSI DIP IC's with busses for power supply connector. $4.00

MV 5691 YELLOW-GREEN BIPOLAR LED .... $ .90
FP 100 PHOTO TRANS .... $ .50
RED, YELLOW, GREEN or AMBER LARGE LED's .... 6/$1.00
1L-5 (MCT-2) .... $ .75
MOLEX PINS .... 100/$1.00; 1000/$8.00
10 WATT ZENERS 3.9, 4.7, 5.6, 8.2, 12, 18, 22, 100, 150 or 200V. . ea. $ .60
1 WATT ZENERS 4.7, 5.6, 10, 12, 15 18 or 22V .... ea. $ .25
MC6860 MODEM CHIP $9.95

Silicon Power Rectifiers

PRV	1A	3A	12A	50A	125A
100	.06	.14	.30	.80	3.70
200	.07	.20	.35	1.15	4.25
400	.09	.25	.50	1.40	6.50
600	.11	.30	.70	1.80	8.50
800	.15	.35	.90	2.30	10.50
1000	.20	.45	1.10	2.75	12.50

SILICON SOLAR CELLS

2¼″ diameter
.4V at 500 ma. $4.00

REGULATORS

309K ...... $ .95
723 ...... $ .50
LM 376 .... $ .60
320K-5 or 15V $1.40
320T-5, 15 or 24V .. $1.15
340K-5, 12, 15 or 24V. .... $ .85
340T-5, 6, 8, 12 15, 18 or 24V $1.10
78 MG $1.35
79 MG $1.35

RS232 CONNECTORS DB 25P male ..... $2.95; DB 25S female $3.50

TANTULUM CAPACITORS

22UF 35V	5/$1.00	10UF 10V	$ .25
47UF 35V	5/$1.00	22UF 25V	$ .40
68UF 35V	5/$1.00	15UF 35V	3/$1.00
1UF 35V	5/$1.00	30UF 6V	5/$1.00
2.2 UF 20V	5/$1.00	33UF 35V	$ .40
3.3UF 35V	4/$1.00	47UF 20V	$ .35
4.7UF 15V	5/$1.00	68 UF 15V	$ .50
6.8UF 35V	4/$1.00		

M7001 ALARM CLOCK CHIP $5.75

NATIONAL MOS DEVICES

MM1402– 1.75	MM5057– 2.25
MM1403– 1.75	MM5058– 2.75
MM1404– 1.75	MM5060– 2.75
MM5013– 2.50	MM5061– 2.50
MM5016– 2.50	MM5555– 4.75
MM5017– 2.70	MM5556– 4.75
MM5055– 2.25	MM5210– 1.95
MM5056– 2.25	MM5260– 1.75

TTL IC SERIES

7400– .16	7445– .65	74151– .65
7401– .16	7446– .70	74153– .65
7402– .16	7447– .65	74154– 1.00
7403– .16	7448– .70	74155– .70
7404– .20	7450– .15	74157– .65
7405– .20	7472– .32	74161– .85
7406– .25	7473– .32	74163– .80
7407– .25	7474– .32	74164– .95
7408– .20	7475– .45	74165– 1.05
7409– .20	7476– .30	74173– 1.35
7410– .16	7480– .45	74174– .95
7411– .20	7483– .70	74175– .90
7412– .20	7485– .87	74176– .75
7413– .39	7486– .30	74177– .75
7414– .85	7489– 1.85	74180 .65
7416– .25	7490– .45	74181– 2.00
7417– .25	7491– .65	74190– 1.20
7420– .16	7492– .43	74191– 1.20
7425– .25	7493– .45	74192– .83
7426– .25	7494– .70	74193– .83
7427– .25	7495– .65	74194– .85
7430– .16	7496– .65	74195– .85
7432– .25	74107– .28	74196– .86
7437– .24	74121– .33	74279– .87
7438– .24	74123– .65	75491– .50
7440– .16	74125– .40	75492– .50
7441– .70	74126– .40	
7442– .52	74150– 1.10	

MINIATURE DIP SWITCHES

CTS-206-8 Eight SPST switches in a 16 pin DIP package. $2.20

5-8V SPST Miniature reed, relay., normally open, 330 Ohm coil resistance. $.75, 3/$2.00

ALCO MINIATURE TOGGLE SWITCHES
MTA 106 SPDT $1.20
MTA 206 DPDT $1.70

Full Wave Bridges

PRV	2A	6A	25A
100			1.30
200	.75	1.25	2.00
400	.95	1.50	3.00
600	1.20	1.75	4.00

SANKEN AUDIO POWER AMPS

Si 1010 G 10 WATTS .... $ 7.95
Si 1020 G 20 WATTS $15.95
Si 1050 G 50 WATTS $27.95

WSU-30-Hand wire wrap tool used to wrap, unwrap & Strip #30 wire $5.30

#24, EIGHT CONDUCTOR SPECTRA FLAT CABLE 10′/$1.50 100′/$13.50

DIP SOCKETS

8 PIN – .22
14 PIN – .25
16 PIN – .28
18 PIN – .30
24 PIN – .40
28 PIN – .50
40 PIN – .60

#30 WIRE WRAP WIRE SINGLE STRAND 100′/$1.40

74LS SERIES / LINEAR CIRCUITS

74LS SERIES		LINEAR CIRCUITS
	74LS125 – .65	
74LS00 – .26	74LS126 – .65	LM 101 – .75
74LS02 – .26	74LS132 – .80	LM 301/748 .29
74LS03 – .26	74LS136 – .39	LM307 – .30
74LS04 – .28	74LS138 – .72	LM 308 – .75
74LS05 – .28	74LS139 – .72	LM 311 – .75
74LS08 – .28	74LS151 – .98	LM 318 –1.20
74LS09 – .28	74LS153 – .93	LM 319 – .95
74LS10 – .26	74LS155 – 2.00	LM 324 –1.05
74LS11 – .26	74LS156 – .95	LM 339 –1.10
74LS13 – .50	74LS157 – .98	LM 358 –1.40
74LS15 – .28	74LS160 –1.02	LM 370 –1.15
74LS20 – .26	74LS161 –1.02	LM 377 –2.50
74LS21 – .26	74LS162 –1.02	LM 380 – .95
74LS22 – .26	74LS163 –1.02	LM 381 –1.25
74LS26 – .33	74LS168 –1.10	LM 382 –1.25
74LS27 – .33	74LS169 –1.10	LM 537 –2.50
74LS30 – .26	74LS170 –1.72	LM 553 –2.50
74LS32 – .33	74LS173 –1.39	LM 555 – .44
74LS37 – .37	74LS174 –1.05	LM556 – .85
74LS38 – .37	74LS175 –1.22	NE540L –2.75
74LS40 – .27	74LS190 –1.50	560 –2.00
74LS42 – .88	74LS191 –1.50	565 –1.10
74LS47 – .79	74LS192 –1.75	566 –1.25
74LS51 – .26	74LS193 –1.75	567 –1.30
74LS54 – .26	74LS195 –1.25	703 – .90
74LS90 – .95	74LS196 – .99	709 – .25
74LS73 – .40	74LS197 – .99	710 – .35
74LS74 – .40	74LS221 –1.25	741C or V – .31
74LS76 – .40	74LS257 –1.35	747 – .65
74LS90 – .89	74LS258 –1.38	LM 1310 2.50
74LS92 – .85	74LS279 – .75	1456 – .95
74LS93 – .85	74LS365 – .66	1458 – .60
74LS109 – .43	74LS366 – .66	CA3046 – .75
74LS112 – .43	74LS367 – .66	3900 – .49
74LS113 – .43	74LS368 – .66	8038CC –3.90
74LS114 – .43	74LS390 – 1.95	UA791P5 – 1.95

TRIACS / SCR'S

PRV	TRIACS 1A	TRIACS 10A	TRIACS 25A	SCR'S 1.5A	SCR'S 6A	SCR'S 35A
100	.40	.70	1.30	.40	.50	1.20
200	.70	1.10	1.75	.60	.70	1.60
400	1.10	1.60	2.60	1.00	1.20	2.20
600	1.70	2.30	3.60		1.50	3.00

Terms: FOB Cambridge, Mass. Send Check or Money Order. Include Postage, Minimum Order $5.00, COD'S $20.00

Send 25¢ for our catalog featuring Transistors and Rectifiers
145 Hampshire St., Cambridge, Mass.

SOLID STATE SALES
P.O. BOX 74B
SOMERVILLE, MASS. 02143 TEL. (617) 547-4005

WE SHIP OVER 95% OF OUR ORDERS THE DAY WE RECEIVE THEM

Reader Service

To get further information on the products advertised in BYTE, fill out the reader service card with your name and address. Then circle the appropriate numbers for the advertisers you select from the list. Add a 9 cent stamp to the card, then drop it in the mail. Not only do you gain information, but our advertisers are encouraged to use the marketplace provided by BYTE. This helps us bring you a bigger BYTE.

Inquiry No.	Page No.
1	ALF 150
2	Alpha Digital 153
3	Anderson Jacobson 51
4	Apple Computer 16, 17, 18
5	Artec 63
134	AVATAR 189
7	Atwood Enterprises 197
8	Base 2 59
9	Beckian Enterprises 199
10	Benson & Costello 191
11	Beta Business Sys 189
12	BITS 123, 125, 160, 167, 192
*	BYTE Back Issues 192
*	BYTE Binders 191
*	BYTE Index 192
14	Byte Shop of Miami 189
15	Byte Shop, Tustin 138
16	Byte Shop, Westminster 149
*	BYTE Subs 127
18	Canada Systems 178
19	Capital Equipment Brokers 176
138	Celetron 133
20	CMC Marketing 57
21	Component Sales 193
22	Computalker 129
23	Computer Corner 189
24	Computer Creations 191
25	Computer Data Sys 105
26	Computer Depot 191
27	Computer Depot Dist 187
28	Computer Enterprises 45
29	Computer Hobbyist 176
30	Computerland 20, 21
31	Computer Machine Service 189
32	Computer Room, San Jose 174
33	Computer Systems Store 189
34	Cromemco 1, 2
35	DaJen 129
36	Data General 143
37	Data Search 179
38	Digital Group 93
39	Digital Research 55
40	Disc 3 187
41	Don Alan Enterprises 187
42	DRC Engineering 201
43	Dynabyte 111
44	Ed Smith's Software Works 182
45	Educational Data Sys 183
46	E & L Instruments 109

Inquiry No.	Page No.
47	Elcom Industries 189, 191, 187
48	Electrolabs 199
49	Elec Control Technology 159
50	Electronic Systems 202
51	Electronics Warehouse Inc 203
62	C W Engel 175
52	Extensys 7
53	Formula International 205
54	Gallagher Research Inc 172
55	General Computer 175
56	Godbout 207
57	DC Hayes 181
58	Heath 34, CIV
59	Heuristics 174
61	IMSAI 70, 71, 73
63	Integral Data Sys 147
64	Integrated Circuits Unlim 209
65	International Data Sys 152
137	Ithaca Audio 173
99	IOR 187
66	Jade 212, 213
67	James 10, 11, 214, 215
68	Kent-Moore 41, 77
69	Logical Services 148
70	Meca 103
71	Meshna 217
72	Micro Diversions 191
136	Micro Ware 158
73	Micronics 179
74	Microsoft 177
75	Midwestern Scientific Instruments 121
76	Mikos 208
77	Mini Micro Mart 172
78	Morrow's Micro Stuff 219
79	Mountain Hardware 146
80	mpi 160
81	MSD 97
17	Mullen 177
82	National Multiplex 81
83	North Star Computers 5, 15
84	Objective Design 185
85	OK Machine & Tool 75
*	Ohio Scientific Research 30, 31, 32, 33
86	Owens Associates 187
87	PAIA 159
88	Parasitic Engineering 22, 23
89	Peninsula Marketing 151
90	PerCom Data 19, 155

Inquiry No.	Page No.
91	Peripheral Vision 67
92	Perri-White 190
93	Personal Computer News 182
94	Phone I 187
*	PolyMorphic Systems 37
95	Processor Technology 86, 87, 88, 89, 90
96	Quantronics 39, 47
97	Rotundra Cybernetics 191
139	S-100 150
98	Scelbi 95, 119, CIII
*	Scientific Research 79, 115
100	SD Sales 221
101	Seals 65
135	Sharp & Associates 211
102	Silver Spur 187
103	Small Computer Sys 173
104	Smoke Signal 52, 53
105	Software Exchange 180
13	Software Records 161
140	Software Unlimited Ltd 158
106	Solid State Music 9
107	Solid State Sales 223
108	Soroc 29
109	SwTPC CII
110	Sunny Computer Stores 191
111	Sybex 183
112	Synchro Sound 82, 83
113	Szerlip 189
114	Tandy 85
115	Tarbell 178
116	Tech Mart 107
117	Technical Design Labs 27
118	Technical Systems Cnsltnts 61
119	Technico 134, 135
20	TEI 57
120	TK Electronics 189
121	TLF 49
122	Typetronics 25
123	μP Electronics 187
124	Urban Instruments 211
125	Vamp 191
126	Vandenberg Data 180
127	Vector Graphic 39
*	Vista Computers 144, 145
129	West Coast Computer Faire 194, 195
130	Worldwide 185
131	WWW Enterprises 161
132	Ximedia 117, 132
133	Xybek 181

**Reader Service inquiries not solicited. Correspond directly with company.*

BOMB — BYTE's Ongoing Monitor Box

Article No.	ARTICLE	PAGE
1	**Schmucker-Tarr: The Computers of Star Trek**	12
2	**Rampil: A Floppy Disk Tutorial**	24
3	**Struve: A $19 Music Interface**	48
4	**Ciarcia: Try an 8 Channel DVM Cocktail**	76
5	**Grappel-Hemenway: Jack and the Machine Debug**	91
6	**Higgins: Structured Programming with Warnier-Orr Diagrams**	104
7	**Smith: Simulation of Motion: Part 2**	112
8	**Wier: A Little Bit on Interrupts**	118
9	**Wenzlaff: Using the PolyMorphics Video Interface**	130
10	**Lahasky: Multiprogramming Simplified**	140
11	**Libes: Where to Get Bargains in Used Computer Equipment**	154
12	**McGath: A Look at LISP**	156
13	**Gaskell: Relative Addressing for the 8080**	162
14	**McGahee: Save Software: Use a UART for Serial IO**	164
15	**Lahore: A User's Report on the Intercept Jr**	186

BOMB Analysis for the September 1977 Issue:

		Rank (Multiples of σ)
1st:	Ciarcia, "Control the World," page 30	2.17
2nd:	Jacoby, "Walsh Functions," page 190	1.14

The standard deviation of this sample was 19% of the mean rating.

The first place winner in our BOMB tally receives a $100 bonus check, and a $50 bonus is forwarded to the second place winner. Be sure to indicate your reactions to this month's issue by rating each article on the BOMB evaluation card and forwarding it to our office. The BOMB card is your direct line to the editors' desks.■

SANTA and SCELBI HAVE A UNIQUE GIFT IDEA FOR EVERYONE ON YOUR LIST!

Scelbi Software Books

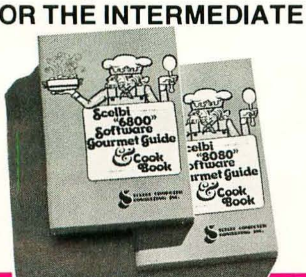
ORDER NOW! SAVE! BEFORE 1978 PRICES GO UP!

FOR THE BEGINNER . . .



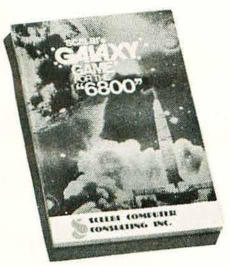
Understanding MicroComputers and Small Computer Systems. A profusely illustrated, easy-reading "must" book explaining fundamental concepts behind operation of microcomputers. Simple English. Gives extra knowledge to read and understand computer magazines and manufacturers' literature. Makes you feel "at home" around computers. Accepted as the standard for the neophyte, you must own this 300-page no-nonsense, easy-reading text. Includes easy-to-use glossary of key microcomputer oriented words. Order now. Save! **$9.95 each ppd.**

FOR THE INTERMEDIATE . . .



Scelbi's Software Gourmet Guides and Cookbooks for '8080' or '6800' lets you cook up mouthwatering programs. Delectable "how to" facts, including '8080' or '6800' instruction sets. How to manipulate stacks. Flow charts. Source listings. General purpose routines for multiple precision operation. Programming time delays for real time. And lots more. Even floating point arithmetic routines! Order your copies today. Bon appetite! Specify: '8080' or '6800'. **$9.95 each ppd.**

FOR THE ADVANCED . . .



GALAXY Microcomputer Outer Space War Games for '6800'. Captain your own starship on intergallactic journeys filled with battles, refueling problems, weaponry, warp factors, and more — all against your '6800'. A complete book, written in machine language for 4K memory. Ever-changing interstellar adventure, includes source listings, flow charts, routines, more. Order today. Blast off tomorrow! **$9.95 ppd.**



SCELBAL. Higher Level Language for '8008'/'8080' Systems. Complete, illustrated program book. Routines. Techniques. Source listings. Flow charts. More. Includes 5 commands, 14 statements, 7 functions, and it runs in 8K and more. All you need to customize a high level language for your system at a fraction of the cost. Order today! **$49 ppd.**



The '8080' Programmer's Pocket Guide; '8080' Octal Code Card and/or '8080' Hexadecimal Code Card. Compact pocket guide for instant reference to either code card. Cards are instant slide rule aids for programming/debugging '8080' software. Standard mnemonics with corresponding codes. Color coded instructions indicate which flags are affected during instruction execution. Quick, logical reference formats. ASCII code chart for 128 characters. '8080' status words. Register pair codes. More. Order all three now . . . **only $2.95 per item.**

Prices shown for North American customers. Master Charge. Postal and bank Money Orders preferred. Personal checks delay shipping up to 4 weeks. Pricing, specifications, availability subject to change without notice.

SCELBI COMPUTER CONSULTING INC.
Post Office Box 133 PP STN
Milford, CT 06460 Dept. B







Get up and running right!...

with the Heathkit H8 8080 Personal Computing System!

H8 8080A 8-Bit Computer	$375
H8-1 4K Memory	140
H8-3 4K Chip Set	95
H8-5 Serial I/O and Cassette Interface	110
H9 Video Terminal	530
ECP-3801 Cassette Recorder/Player	60
If purchased separately,	**$1310.00**
Heath System Price	**$1244.50**

NEW! Microprocessor Course AND TRAINER

Learn the operation and programming of microprocessors with our effective self-study course and trainer!

When you invest in a personal computing system, you want it to perform. So you need software. But putting a complete hardware-software system together can be difficult. Especially if all of the components are not designed to work together. The H8 computer, software and peripherals were developed as a total system. And we include the software (at NO extra cost) so you can start programming right away. Benton Harbor BASIC with unique statements and commands and efficient compression techniques lets you put more program in less space. The HASL-8 2-pass assembler generates efficient machine language code. The TED-8 line-oriented text editor and BUG-8 terminal console debugging program permit fast entry, editing and debugging of programs. The H8's intelligent front panel provides efficient one-button program loading, and lets you "see" what's going on in the machine — in memory, in registers and at I/O ports.

While the H8 and its software gets you up and running, our peripherals complete the "system". The H9 12" CRT video terminal is complete with ASCII 67-key keyboard, long and short-form displays, auto-scrolling, plot mode, cursor and more. And our ECP-3801 cassette recorder/player provides convenient mass storage for all your programs on easy-to-handle cassettes.

All this, plus complete documentation, service support through the Heathkit factory or Heathkit Electronic Centers nationwide*, and self-instructional programming courses make Heath your BEST choice for a truly practical and versatile computer system that's ready and waiting for your commands.

HEATH Schlumberger

Send for your Heathkit Catalog or visit your Heathkit Electronic Center.

***See listing page 34**

Circle 58 on inquiry card.